User's Manual

RENESAS

RZ/G2L Group, RZ/G2LC Group

User's Manual: Hardware

Renesas Microprocessor
RZ Family / RZ/G Series

arm

Renesas Electronics
www.renesas.com

Rev.1.45 Jan, 2024

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Trademarks (continued)

For the “Cortex” notation, it is used as follows;
— Arm® Cortex®-A55
— Arm® Cortex®-M33
Note that after this page, they may be noted as Cortex-A55 and Cortex-M33 respectively.

Examples of trademark or registered trademark used in the RZ/G series, 2nd Generation User’s Manual: Hardware;
AMBA®: AMBA is a registered trademark of Arm Limited.
CoreLink™: CoreLink is a trademark of Arm Limited.
CoreSight™: CoreSight is a trademark of Arm Limited.
Mali™: Mali is a trademark of Arm Limited.
TrustZone ®: TrustZone is a registered trademark of Arm Limited.
HyperFlash™: HyperFlash is a trademark of Cypress Semiconductor Corporation.
MIPI®: MIPI is a registered trademark of MIPI Alliance, Inc.
Neon™: Neon is a trademark of Arm Limited.
OpenGL ES™: OpenGL ES is a trademark of Khronos Group, Inc.
DynamIQ™: DynamIQ is a trademark of Arm Limited.
OctaFlash™: OctaFlash is a trademark of Macronix International Co., Ltd.
eMMC™: eMMC is a trademark of MultiMediaCard Association.

Note that in each section of the Manual, trademark notation of ® and TM may be omitted.
All other trademarks and registered trademarks are the property of their respective owners.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Table of Contents

6. System Controller (SYSC) 165

7. Clock Pulse Generator (CPG) 233

18. General PWM Timer (GPT)....1024

1. Overview

1.1 Introduction

The RZ/G2L and RZ/G2LC includes:

- RZ/G2L RZ/G2LC
  - 1.2 GHz Arm® Cortex®-A55 Dual / Single MPCore cores,
  - 200-MHz Arm® Cortex®-M33 core,
  - 500-MHz Arm® Mali™-G31,
  - Memory controller for DDR4-1600 / DDR3L-1333 with 16 bits,
  - Video processing unit,
  - USB2.0 host / function interface,
  - Gigabit Ethernet interface,
  - SD card host interface,
  - CAN interface,
  - Sound interface.

- RZ/G2L
  - 1 channel MIPI DSI interface or 1channel parallel output interface selectable,
  - 1 channel MIPI CSI-2 input interface or 1channel parallel input interface selectable

- RZ/G2LC
  - 1 channel MIPI DSI interface,
  - 1 channel MIPI CSI-2 input interface

NOTE

Arm and Cortex are registered trademark of Arm Limited. All other brands or product names are the property of their respective holders.

1.2 List of Specifications

1.2.1 CPU Core

Item	Description
System CPU Cortex-A55	[RZ/G2L RZ/G2LC] ● Arm Cortex-A55 Dual / Single MPCore 1.2 GHz ● L1 I-cache 32 Kbytes (Parity) / D-cache 32 Kbytes (ECC) ● L2 cache 0 Kbyte ● L3 cache 256 Kbytes (ECC) ● Arm® NEON™ / FPU supported ● Cryptographic Extension supported ● Arm® v8.2-A architecture
System CPU Cortex-M33	[RZ/G2L RZ/G2LC] ● Arm Cortex-M33 Processor 200 MHz ● Security Extension supported ● Arm® v8-M architecture
Boot	[RZ/G2L RZ/G2LC] ● 6 boot modes Boot Mode 0: Booting from eSD Boot Mode 1: Booting from eMMC (1.8 V) Boot Mode 2: Booting from eMMC (3.3 V) Boot Mode 3: Booting from a serial flash memory (Single / Quad / Octal) connected to the SPI Multi I/O bus space (1.8 V) Boot Mode 4: Booting from a serial flash memory (Single / Quad) connected to the SPI Multi I/O bus space (3.3 V) Boot Mode 5: Booting from SCIF download
Debug Interface	[RZ/G2L RZ/G2LC] ● Arm® CoreSight™ architecture ● JTAG / SWD interface supported ● ETF 16 Kbytes for program flow trace (each cluster) ● JTAG Disable supported

1.2.2 CPU Peripheral

Item	Description
Clock Pulse Generator (CPG)	[RZ/G2L RZ/G2LC] ● Generates the clocks from external clock (EXCLK 24 MHz). Maximum Arm Cortex-A55 clock: 1.2 GHz Maximum Arm Cortex-M33 clock: 200 MHz Maximum DDR clock: 666 MHz (DDR3L-1333), 800 MHz (DDR4-1600) Maximum 3DGE clock: 500 MHz Maximum VCP clock: 200 MHz Maximum AXI-bus clock: 200 MHz Maximum APB-bus clock: 100 MHz ● SSC (Spread Spectrum Clock) supported
Direct Memory Access Controller (DMAC)	[RZ/G2L RZ/G2LC] ● 2 modules, 16 channels per module ● Transfer request: On-chip peripheral request / auto request (software trigger) ● A specific DMA transfer interval can be specified to adjust the bus occupancy. ● LINK mode (DMA transfer under descriptor control) supported ● Transfer information can be automatically reloaded
Interrupt Controller	[RZ/G2L RZ/G2LC] ● Arm® CoreLink™ Generic Interrupt Controller (GIC-600) for Arm Cortex-A55 ● Nested Vectored Interrupt Controller (NVIC) for Arm Cortex-M33 ● External Interrupt pins (NMI, IRQ7 to IRQ0*1, TINT31-0) ● On-chip peripheral Interrupts: Priority level set for each module *Note 1.* [RZ/G2LC] IRQ0,1,4-7 are available.
Message Handling Unit (MHU)	[RZ/G2L RZ/G2LC] ● Message handling function between Arm Cortex-A55 and Arm Cortex-M33 ● Assert interrupt to inform message and response from/to Arm Cortex-A55, Cortex-M33
General-purpose I/O (GPIO)	[RZ/G2L RZ/G2LC] ● General-purpose I/O ports
Thermal Sensor Unit (TSU)	[RZ/G2L RZ/G2LC] ● 1 channel

1.2.3 Internal Memory

Item	Description
On-chip RAM	[RZ/G2L RZ/G2LC] ● RAM of 128 Kbytes (ECC)

1.2.4 External Memory Interface

Item	Description
External Bus Controller for DDR3L / DDR4 SDRAM (DDR)	[RZ/G2L RZ/G2LC] ● Support DDR3L-1333 / DDR4-1600 ● Bus Width: 16-bit ● In line ECC supported (Support error detection interrupt) ● Memory Size: Up to 4 Gbyes ● Auto Refresh supported
SPI Multi I/O Bus Controller	[RZ/G2L] ● 1 channel (8-bit Double data rate) ● Up to 2 serial flash memories with multiple I/O bus sizes (single / quad) can be connected ● Connectable with 1 Octal-SPI flash memory ● Connectable with 1 HyperFlash memory ● External address space read mode (built-in read cache) ● SPI operation mode ● Maximum Clock Frequency: – 50 MHz (Quad-SPI DDR) – 66 MHz (Quad-SPI SDR) – 100 MHz (Octal-SPI, HyperFlash) [RZ/G2LC] ● 1 channel (4-bit Double data rate) ● 1 serial flash memories with multiple I/O bus sizes (single / quad) can be connected ● External address space read mode (built-in read cache) ● SPI operation mode ● Maximum Clock Frequency: 50 MHz (Quad-SPI DDR), 66 MHz (Quad-SPI SDR)
SD Card Host Interface / Multimedia Card Interface (SD/MMC)	[RZ/G2L RZ/G2LC] ● 2 channels ● Channel 0 supports SDHI / e-MMC (boot supported) ● Channel 1 supports SDHI ● SD memory I/O card interface (1-bit / 4-bit SD bus) ● SD, SDHC and SDXC SD memory card access supported ● Compliant with SD 3.0 ● Default, high-speed, UHS-I/SDR50, SDR104 transfer modes supported ● Error check function: CRC7 (Command/response), CRC16 (Data) ● Card detection function, write protect supported ● MMC interface (1-bit / 4-bit / 8-bit MMC bus) ● e-MMC device access supported ● Compliant with eMMC 4.51 ● High-speed, HS200 transfer modes supported

1.2.5 Graphics Unit

Item	Description
3D Graphics Engine (3DGE)	[RZ/G2L RZ/G2LC] ● Arm Mali-G31 ● One single-pixel shader core ● 8 Kbytes L2 Cache ● OpenGL ES1.1 / 2.0 / 3.0 / 3.1 and 3.2 Supported ● OpenCL 2.0 Full Profile Supported

1.2.6 Video Processing Unit

Item	Description
Video Codec Processor (VCP)	[RZ/G2L] ● H.264 codec module ● Encoding / Decoding support – H.264 / AVC (High Profile / Main Profile / Baseline Profile) – H.264 / MVC (Stereo High Profile) ● Maximum pixel rate: 1920 × 1080 × 30 fps ● Color format (input in encoding): YcbCr 4:2:0 semi-planar supported ● Color format (output in decoding): YcbCr 4:2:0 semi-planar supported
Image Scaling Unit (ISU)	[RZ/G2L RZ/G2LC] ● Scaling down function with bilinear interpolation ● Input image Size (max): 5M (2800 × 2047) ● Output image Size (max): Full HD (1920 × 1080) ● Support Color format Conversion ● RGB / ARGB / YcbCr422 / YcbCr420 / RAW (Grayscale)

1.2.7 Camera Interfaces

Item	Description
MIPI CSI-2 Interface	[RZ/G2L RZ/G2LC] ● 1 channel ● The number of Lane: 1-/2-/4-lane ● Support 5MP, 30 fps (RAW12) ● Maximum Bandwidth: 1.5 Gbps per lane ● Select 1 VC from 4 VC (virtual channel) supported ● Support Input Image Data Formats: – YUV420 8-bit / 10-bit – Legacy YUV420 8-bit – YUV420 8-bit / 10-bit (Chroma Shifted Pixel Sampling) – YUV422 8-bit / 10-bit – RGB444 / RGB555 / RGB565/ RGB666 / RGB888 – RAW6 / RAW7 / RAW8 / RAW10 / RAW12 / RAW14 / RAW16 / RAW20 ● Generic short packet code 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8 ● Generic long packet data type 1 / 2 / 3 / 4 ● User Defined 8-bit data type 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8
Parallel Input Interface	[RZ/G2L] ● 1 channel ● Support ITU-R BT.656 Interface (Interlace supported, YcbCr422 8-bit / 10-bit) ● Support HD – 30 fps (YCbCr422 Interleave), 60 fps (YCbCr422 Y/CbCr separate data, binary data) – Maximum input pixel frequency: 108 MHz ● Support Input Data Format: – YcbCr422 8-bit / 10-bit – Binary data 16-bit ● VSYNC / HSYNC / FIELD timing signal supported

Item	Description
RZ/G2L MIPI CSI-2 / Parallel to AXI Bridge Module RZ/G2LC MIPI CSI-2 to AXI Bridge Module	[RZ/G2L] ● 1 channel (MIPI CSI-2 Input or Parallel Input) ● Support Image Processing: – Clipping – Frame Sampling – LUT – Color format conversion – Color space conversion – Demosaicing – Linear Matirix – Statistics ● Support Color Formats for Image Processing: – YUV422 8-/10-bit – RGB565 / RGB666 / RGB888 – RAW8 / 10 / 12 / 14 / 16 (Clipping and Frame Sampling only) ● Support Output Data Formats: – YCbCr422 8-bit (Interleave/Semi planar, Interlace/Progressive) – YCbCr420 8-bit (Interleave, Interlace) – Y-Only – RGB888 / ARGB8888 – RAW8 / 10 / 12 / 14 / 16 (without Image Processing) – MIPI CSI-2 V2.1 Recommended Memory storage data (without Image Processing) [RZ/G2LC] ● 1 channel (MIPI CSI-2 Input) ● Support Image Processing: – Clipping – Frame Sampling – LUT – Color format conversion – Color space conversion – Demosaicing – Linear Matirix – Statistics ● Support Color Formats for Image Processing: – YUV422 8/10-bit – RGB565 / RGB666 / RGB888 – RAW8 / 10 / 12 / 14 / 16 (Clipping and Frame Sampling only) ● Support Output Data Formats: – Y-Only – RGB888 / ARGB8888 – RAW8/10/12/14/16 (without Image Processing) – MIPI CSI-2 V2.1 Recommended Memory storage data (without Image Processing)

1.2.8 Display Interface

Item	Description
LCD Controller	[RZ/G2L] ● 1 channel (MIPI DSI output or Parallel output) [RZ/G2LC] ● 1 channel (MIPI DSI output) [RZ/G2L RZ/G2LC] ● 2 planes blending (can blend 2 different size images) ● Support Image Processing: – Dither processing (RGB666) – Clipping – RGB Gamma Correction LUT ● Support Input Data Format: – RGB565 / RGB666 / RGB888 – ARGB1555 / ARGB4444 / ARGB8888 – YcbCr444 8-bit / YcbCr422 8-bit / YcbCr420 8-bit
MIPI DSI Interface	[RZ/G2L RZ/G2LC] ● 1 channel ● The number of Lane: 4-lane ● Support up to Full HD (1920 × 1080), 60 fps (RGB888) ● Maximum Bandwidth: 1.5 Gbps per lane ● Support Output Data Format: – RGB666 / RGB888
Parallel Output Interface	[RZ/G2L] ● 1 channel ● Support WXGA (1280 × 800), 60 fps ● Support Output Data Format: – RGB666 / RGB888 ● CLK / HD / VD timing signal supported

1.2.9 Sound Interface

Item	Description
Serial Sound Interface (SSI)	[RZ/G2L] ● 4 channels bidirectional serial transfer [RZ/G2LC] ● 3 channels bidirectional serial transfer [RZ/G2L RZ/G2LC] ● 2 external clock sources available ● Full Duplex communication (channel 0, 1, and 3) ● Support of I2S / Monaural / TDM audio formats ● Support of master and slave functions ● Generation of programmable word clock and bit clock ● Multi-channel formats ● Support of 8, 16, 18, 20, 22, 24, and 32-bit data formats ● Support of 32-stage FIFO for transmission and reception ● Support of LR-clock continue function in which the LR-clock signal is not stopped
Sampling Rate Converter (SRC)	[RZ/ G2L RZ/G2LC] ● 1 channel ● Data format: 16-bit (stereo / monaural) ● Sampling Rate Input: Selectable from 8 kHz, 11.025 kHz, 12 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, 48 kHz Output: Selectable from 8 kHz*, 16 kHz*, 32 kHz, 44.1 kHz, 48 kHz (*: can select in 44.1 kHz input mode) ● SNR: More than or equal to 80 db

1.2.10 Storage and Network

Item	Description
USB2.0 Host / Function (USB)	[RZ/G2L RZ/G2LC] ● 2 channels (ch0: Host-Function ch1: Host only) ● Compliance with USB2.0 ● Supports On-The-Go (OTG) Function ● Supports Battery Charging Function ● Internal dedicated DMA
Gigabit Ethernet Interface (GbE)	[RZ/G2L] ● 2 channels [RZ/G2LC] ● 1 channel [RZ/G2L RZ/G2LC] ● Supports transfer at 1000 Mbps and 100 Mbps, 10 Mbps ● Supports filtering of Ethernet frames ● Supports interface conforming to IEEE802.3 PHY RGMII (Reduced Gigabit Media Independent Interface) ● Supports interface conforming to IEEE802.3 PHYMII (Media Independent Interface)
CANFD Interface (RS-CANFD)	[RZ/G2L RZ/G2LC] ● 2 channels ● ISO 11898-1 (2003) compliant ● CAN-FD ISO 11898-1 (CD2014) compliant ● Message buffer – Up to 64 × 2-channel receive message buffer: Shared among all channels – 16 transmit message buffers per channel

1.2.11 Timer

Item	Description
Multi-function Timer Pulse Unit 3 (MTU3a)	[RZ/G2L] ● 9 channels (16 bits × 8 channels, 32 bits × 1 channel) [RZ/G2LC] ● 8 channels (16 bits × 7 channels, 32 bits × 1 channel) [RZ/G2L RZ/G2LC] ● Module clock frequency (P0ϕ): 100 MHz ● Maximum 28 lines of pulse inputs/outputs and 3 lines of pulse inputs ● 14 types of count clocks selectable ● Input capture function ● 39 outputs compare and input capture registers ● Counter clear operation (Simultaneous counter clearing by Compare match or Input capture is available) ● Simultaneous writing to multiple timer counters (TCNT) ● Synchronous input/output of each register due to synchronous operation of the counter ● Buffered operation ● Cascade-connected operation ● 43 types of interrupt sources ● Automatic transfer of register data ● Pulse output modes Toggle, PWM, complementary PWM, and reset-synchronized PWM modes ● Synchronization of multiple counters ● Phase counting mode – 16-bit mode (channel 1 and 2) – 32-bit mode (channel 1 and 2) ● Counter function of dead time compensation ● Digital filter functions for the input capture and external count clock pin
Port Output Enable 3 (POE3)	[RZ/G2L RZ/G2LC] ● Control of the high-impedance state of the MTU3a waveform output pins ● Activation with four input pins ● Activation on detection of short-circuited outputs (detection of simultaneous PWM output to the active level) ● Activation by register write ● Additional programming of output control target pins is possible.

Item	Description
General PWM Timer (GPT)	[RZ/G2L] ● 32 bits × 8 channels [RZ/G2LC] ● 32 bits × 6 channels [RZ/G2L RZ/G2LC] ● Counting up or down (sawtooth wave), counting up and down (triangular wave) selectable for all channels ● Independent selectable for each channel ● 2 input/output pins per channel ● 2 output compare / input capture registers per channel ● For the 2 output compare / input capture registers of each channel, 4 registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use ● In output compare operation, buffer switching can be at peaks or troughs, enabling the generation of laterally asymmetrically PWM waveforms ● Registers for setting up frame intervals on each channel (with capability for generating interrupts on overflow or underflow) ● Generation of dead times in PWM operation ● Synchronous start / stop / clear of counters on arbitrary channels ● Starting, stopping, and clearing up/down counters in response to a maximum of eight events ● Starting, stopping, and clearing up/down counters in response to input level comparison ● Starting, stopping, and clearing up/down counters in response to a maximum of four external triggers ● Output pin invalidation functions due to dead time error or detection of short circuit between output pins ● Digital filter functions for the input capture and external trigger pins
Port Output Enable for GPT (POEG)	[RZ/G2L RZ/G2LC] ● Output prohibition control of the GPT waveform output pin ● Activation with up to four input pins ● Activation by dead time error detection or output short detection ● Activation by register write
Watchdog Timer (WDT)	[RZ/G2L RZ/G2LC] ● 3 channels ● A counter overflow can reset the LSI ● CPU parity error can reset the LSI
General Timer (GTM)	[RZ/G2L RZ/G2LC] ● 32 bits × 3 channels ● Two operating modes – Interval timer mode – Free-running comparison mode

1.2.12 Peripheral Module

Item	Description
I2C Bus Interface (I2C)	[RZ/G2L RZ/G2LC] ● 4 channels ● Master mode and slave mode supported ● Support for 7-bit and 10-bit slave address formats ● Support for multi-master operation ● Timeout detection
Serial Communication Interface with FIFO (SCIFA)	[RZ/G2L] ● 5 channels [RZ/G2LC] ● 4 channels [RZ/G2L RZ/G2LC] ● Clock synchronous mode or asynchronous mode selectable ● Simultaneous transmission and reception (full-duplex communication) supported ● Dedicated baud rate generator ● Separate 16-byte FIFO registers for transmission and reception ● Modem control function (channel 0, 1, and 2 in asynchronous mode)
Serial Communication Interface (SCIg)	[RZ/G2L RZ/G2LC] ● 2 channels ● Clock synchronous mode, asynchronous mode, or smart card interface mode is selectable ● Simultaneous transmission and reception (full-duplex communication) supported ● Dedicated baud rate generator ● LSB first / MSB first selectable ● Modem control function ● Encoding and decoding of IrDA communications waveforms in accord with version 1.0 of the IrDA standard (on channel 0)
Renesas Serial Peripheral Interface (RSPI)	[RZ/G2L RZ/G2LC] ● 3 channels ● SPI operation ● Master mode and slave mode supported ● Programmable bit length, clock polarity, clock phase can be selected ● Consecutive transfers ● LSB first / MSB first selectable

1.2.13 Security

Item	Description
Trusted Secure IP (TSIP) [option]	[RZ/G2L RZ/G2LC] ● Security algorism – Common key encryption: AES – Non-common key encryption: RSA, ECC ● Other features – TRNG (true-random number generator) – Hash value generation: SHA-1, SHA-224, SHA-256, GHASH – Support of Unique ID
One Time Programmable memory (OTP)	[RZ/G2L RZ/G2LC] ● A nonvolatile memory that can be written only once ● Security setting, authentication setting are possible ● Support one time read function (512 bytes)

1.2.14 Analog

Item	Description
A/D Converter (ADC)	[RZ/G2L] ● 8 channels ● Resolution: 12-bit ● Input Range: 0 V ~ 1.8 V ● Conversion Time: 1 µs ● Operation Mode: Select mode / scan mode ● Conversion Mode: Single mode / repeat mode ● Condition for A/D conversion start – Software trigger – Asynchronous trigger: External trigger supported – Synchronous trigger: MTU and PWM timer

1.2.15 Others

Item	Description
Boundary Scan	[RZ/G2L RZ/G2LC] ● Boundary scan based on IEEE 1149.1 via JTAG interface is supported. Note that some module pins are not available on this boundary scan.

1.2.16 Power Supply Voltage

Item	Description
Power supply voltage	[RZ/G2L RZ/G2LC] ● V_{DD}, PLLn_DV_{DD11} (n = 23, 5): 1.05 to 1.15 V ● DDR_V_{DDQ}: 1.14 to 1.26 V (DDR4) / 1.283 to 1.45 V (DDR3L) ● V_{DD18}, ADC_AV_{DD18}, PLLn_AV_{DD18} (n = 1, 23, 4, 5, 6): 1.62 to 1.98 V ● OTP_V_{DD18}, USB_V_{DD18}, CSI_V_{DD18}, DSI_V_{DD18}: 1.65 to 1.95 V ● PV_{DD}: 2.97 to 3.63 V ● USB_V_{DD33}: 3.00 to 3.60 V ● SDn_PV_{DD} (n = 0, 1), SPI_PV_{DD}: 2.97 to 3.63 V / 1.70 to 1.95 V ● $PV_{DD182533}$: 2.97 to 3.63 V / 2.25 to 2.75 V / 1.62 to 1.98 V

1.2.17 Temperature Range

Item	Description
Temperature range	[RZ/G2L RZ/G2LC] ● T_a: −40°C to +85°C*[1] ● T_j: −40°C to +125°C

Note 1. If wider temp is required than this range, use case has to be investigated.

1.2.18 Quality level

Item	Description
Quality level	[RZ/G2L RZ/G2LC] ● Industrial usage, etc.

1.2.19 Package

Item	Description
Package	[RZ/G2L] ● 551-pin BGA, 21-mm square, 0.8-mm pitch ● 456-pin BGA, 15-mm square, 0.5-mm pitch [RZ/G2LC] ● 361-pin BGA, 13-mm square, 0.5-mm pitch

1.3 Product Lineup

Table 1.1 Product Lineup

Group	Package	Part Number	CPU	Security*[1]
RZ/G2L	21 mm BGA	R9A07G044L28GBG	2× Cortex-A55, 1× Cortex-M33	Available
		R9A07G044L18GBG	1× Cortex-A55, 1× Cortex-M33	
		R9A07G044L24GBG	2× Cortex-A55, 1× Cortex-M33	Not supported
		R9A07G044L14GBG	1× Cortex-A55, 1× Cortex-M33	
	15 mm BGA	R9A07G044L27GBG	2× Cortex-A55, 1× Cortex-M33	Available
		R9A07G044L17GBG	1× Cortex-A55, 1× Cortex-M33	
		R9A07G044L23GBG	2× Cortex-A55, 1× Cortex-M33	Not supported
		R9A07G044L13GBG	1× Cortex-A55, 1× Cortex-M33	
RZ/G2LC	13 mm BGA	R9A07G044C26GBG	2× Cortex-A55, 1× Cortex-M33	Available
		R9A07G044C16GBG	1× Cortex-A55, 1× Cortex-M33	
		R9A07G044C22GBG	2× Cortex-A55, 1× Cortex-M33	Not supported
		R9A07G044C12GBG	1× Cortex-A55, 1× Cortex-M33	

Note 1. The product with security function supports the following features.

- — Crypto Extension (Cortex-A55 configuration)
- — Trusted Secure IP
- — Trust Zone
- — Secure Software PKG
- — Secure Boot
- — Secure Debug
- — HW Key protection
- — True Random Generator
- — Trusted Execution Environment (TEE)

Note: RZ/G2LC specification is not written in individual chapters later.
Regarding *RZ/G2LC specification*, see *Section 1, Overview*.

2. System CPU Cortex-A55

The Cortex-A55 system CPU is a core block equipped with single Cortex-A55 core or dual Cortex-A55 cores. For details on the functions including interrupts of the Cortex-A55, see the Arm® Cortex®-A55 Core Technical Reference Manual and related documents in **Section 2.4**. For details of interrupt type and control, see the chapter on Interrupt Controller.

2.1 Features

The Cortex-A55 incorporates an extensively redesigned microarchitecture system that improves performance across the board while being very competitive in area and power efficiency.

Table 2.1 Function Summary

	Description
CPU	● 1.2 GHz Single Cortex-A55 r2p0 (Armv8-A) core or dual Cortex-A55 r2p0 (Armv8-A) cores
Cache memory	● L1$ (I/D) = 32 KB(Parity) / 32 KB (ECC) ● L2$ = None ● L3$ = 256 KB (ECC)

Table 2.2 Core Configuration

	Value	Description
L1 instruction cache size	32 KB	L1 instruction cache = 32 KB (Parity)
L1 data cache size	32 KB	L1 data cache = 32 KB (ECC)
L3 cache size	256 KB	L3 cache = 256KB (ECC), Shared with Dual cores
ECC or parity core cache protection	Included	Support for core cache ECC
Advanced SIMD and floating-point support (including dot product instruction support)	Included	Support for NEON™ FPU engine
Cryptographic extension	Included (option)	Support for cryptography engine
Big endian	N/A	Not supported

2.2 Resets

Table 2.3 shows the supported types of reset and the areas that are reset by the respective types. For the procedures of handling the individual resets, see **Section 2.2.1** and the subsequent parts. See the Arm® DynamIQ™ Shared Unit Technical Reference Manual and Arm® Cortex®-A55 Core Technical Reference Manual.

Table 2.3 Areas to be Reset

Reset Function	Areas to be Reset
Cold reset	Entire area
Cluster warm reset	Entire area of the cluster except for the debug, RAS, and ETM registers + core 0 (and core 1 in dual-core devices)
Core warm reset	Entire area of the target core except for the debug, RAS, and ETM registers
Debug reset	DebugBlock

The P-Channels can be used to control safety in resetting. For details on the P-Channels, see the AMBA® Low Power Interface Specification Arm® Q-Channel and P-Channel Interfaces. For details on registers for controlling the P-Channel and reset pins, see the chapter on the clock pulse generator (CPG). **Table 2.4** shows the supported power modes for each P-Channel. Transitions to power modes other than those listed in the table below are not supported.

Table 2.4 Supported Power Modes

PSTATE	Power Mode
CLUSTERPSTATE[6:0]	● 100_1000b (ON) ● 000_0000b (OFF)
COREPSTATE0[5:0]	● 00_1000b (ON) ● 00_0001b (OFF_EMU) ● 00_0000b (OFF)
COREPSTATE1[5:0]	● 00_1000b (ON) ● 00_0001b (OFF_EMU) ● 00_0000b (OFF)

Reset address vector can be set by below Reset Vector Address Configuration Registers. For details, see the chapter of System Controller.

CA55 Core0 Reset Vector Address High/Low Configuration Register:
SYS_CA55_CFG_RVAH0 / SYS_CA55_CFG_RVAL0

CA55 Core1 Reset Vector Address High/Low Configuration Register:
SYS_CA55_CFG_RVAH1 / SYS_CA55_CFG_RVAL1

To change reset vector address, follow the procedure below.

1. Store codes to new reset address.
2. Write the new reset address to CA55 Reset Vector Address Configuration Registers.
3. Issue reset to target core(s).

2.2.1 Cold Reset

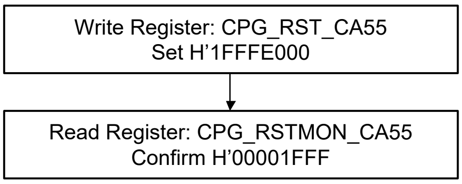
A cold reset is applied at any of the following cases.

(a) The sequence of a power-on reset by the CPG is executed.

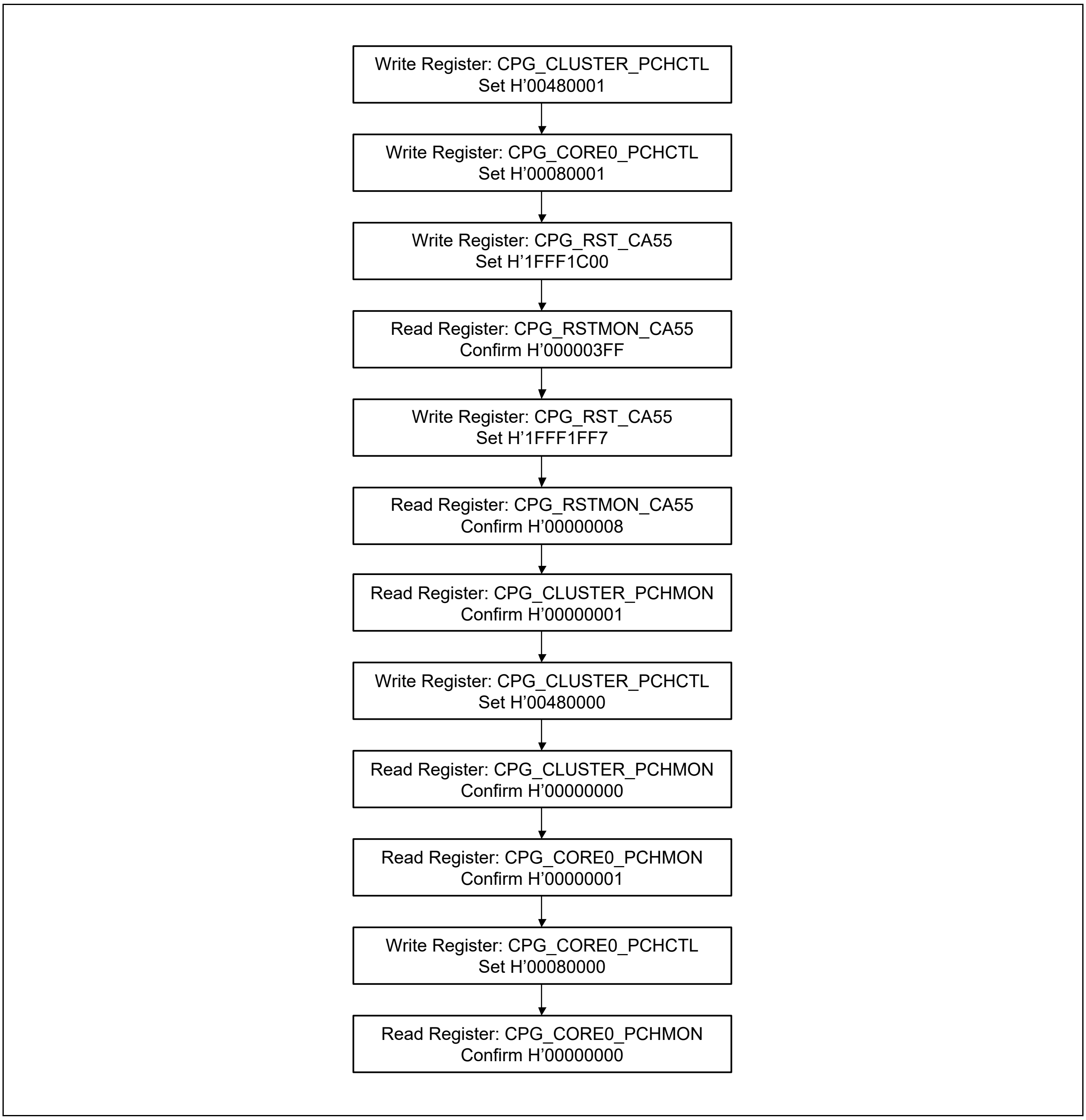
(b) The WDT counter has overflowed or a non-correctable error has been detected.

(c) Software control by the Cortex-M33 system CPU is applied.

For details on (a) and (b) reset, see the chapters on the clock pulse generator (CPG) and watchdog timer (WDT), respectively. To apply a cold reset by case c reset, follow the procedure below.



To release cold reset and start core 0 by case c reset, follow the procedure below.



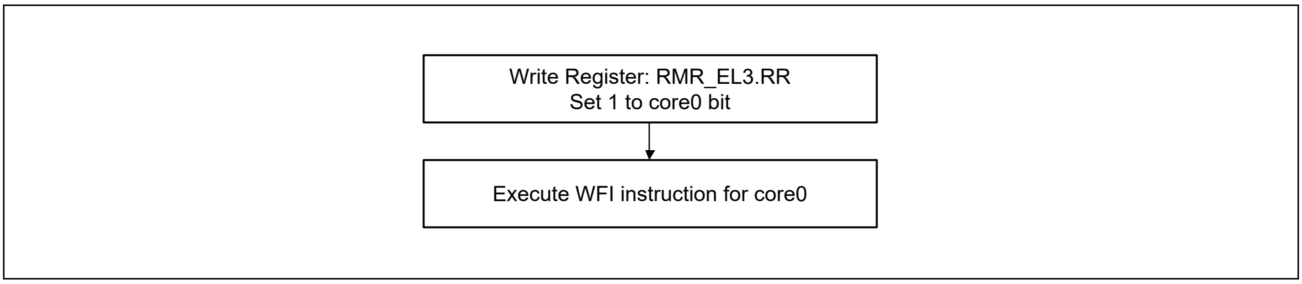
Procedure to start core 1 after having started core 0 is the same as that in the power-on reset sequence. For details, see the chapter on the clock pulse generator (CPG).

2.2.2 Core Warm Reset

There are the two types of core warm reset: internal and external resets. The former requires software control by the Cortex-A55 system CPU itself and the latter requires software control by the Cortex-A55 and Cortex-M33 system CPUs. Though the Cortex-A55 system CPU handles an internal reset in the same way as an external reset, if you intend to reset a single core, we recommend using an internal reset because handling this requires fewer steps than an external reset. An external reset is mainly used as part of a cluster warm reset, which is described in **Section 2.2.3**.

To apply an internal reset to core 0, follow the procedure below.

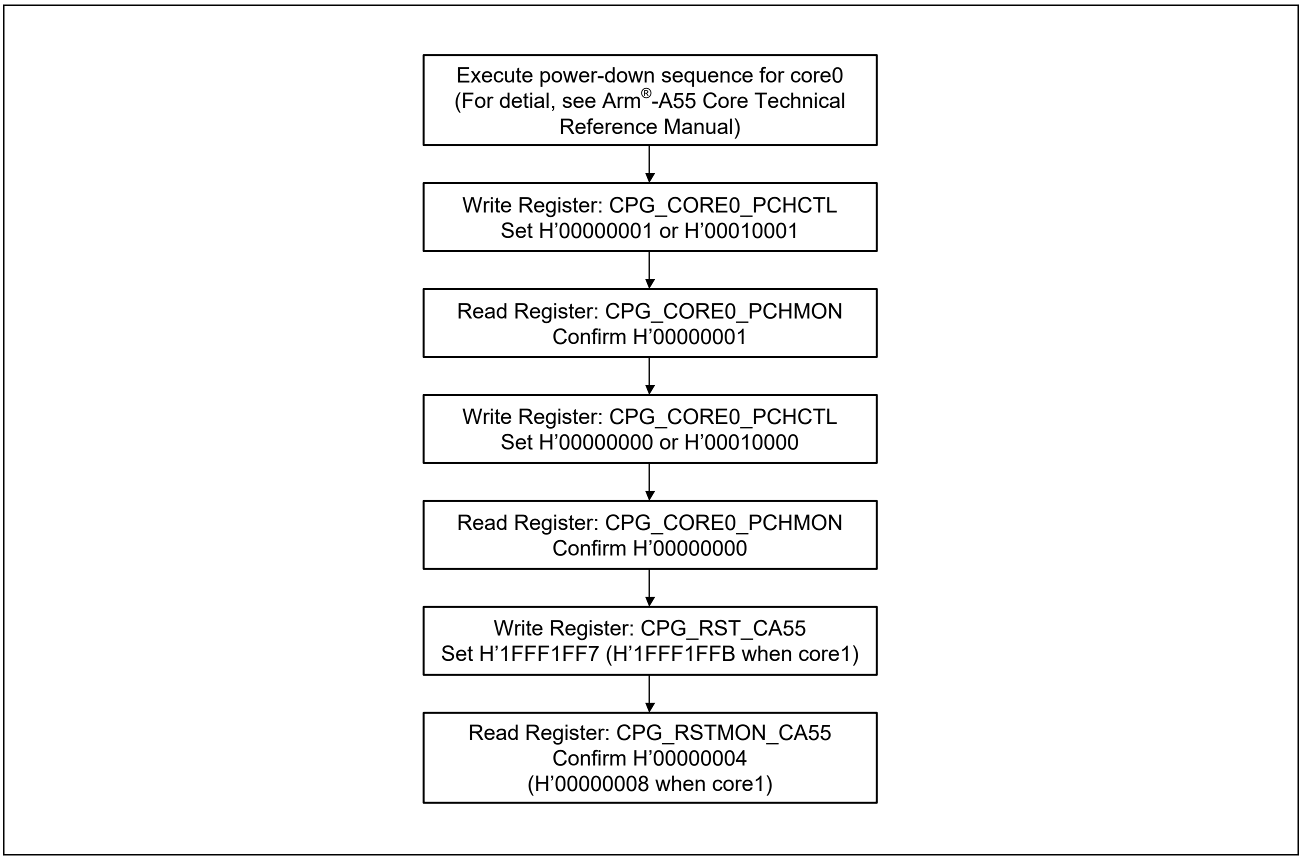
To apply an internal reset to core 1, read “core 0” as “core 1” in the sentences below.



Release from an internal reset is automatic following application of the reset. No specific processing is required. For details, see the Arm® DynamIQ™ Shared Unit Technical Reference Manual and Arm® Cortex®-A55 Core Technical Reference Manual.

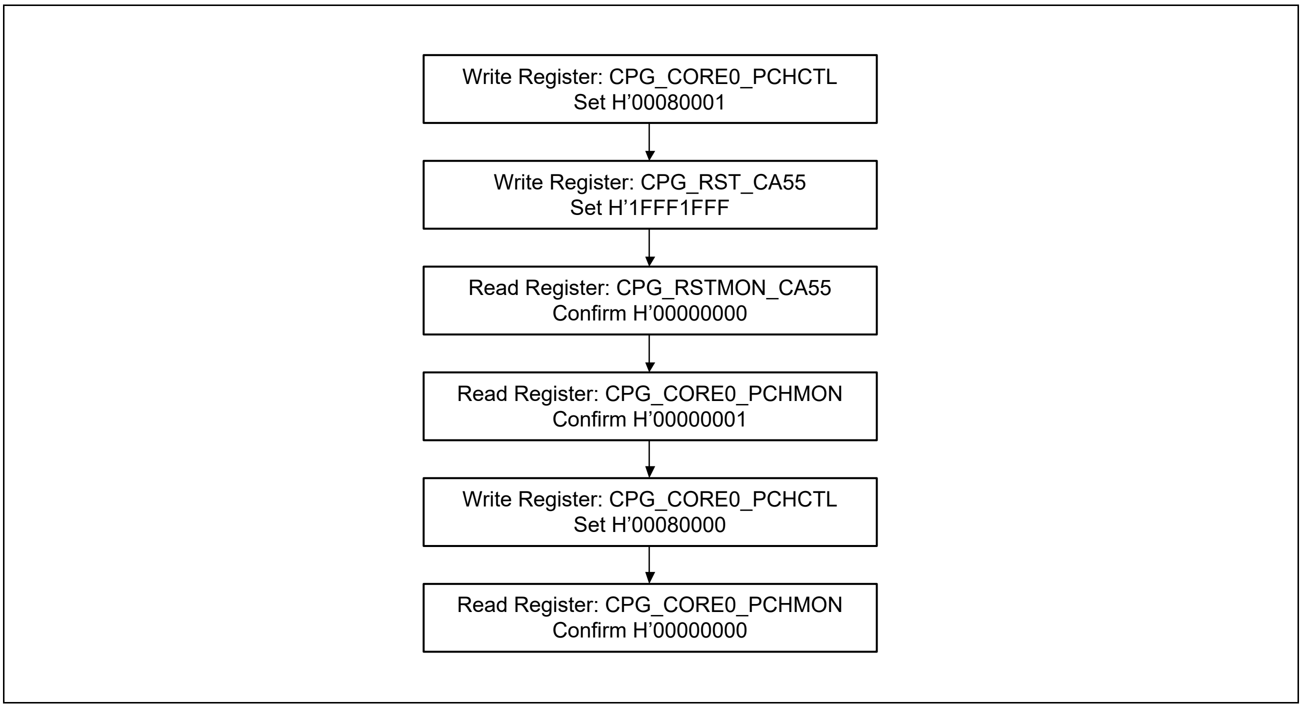
To apply an external reset to core 0, follow the procedure below.

To apply an external reset to core 1, read “core 0” as “core 1” in the sentences below.



To release core 0 from the core warm reset state and start the core, follow the procedure below.

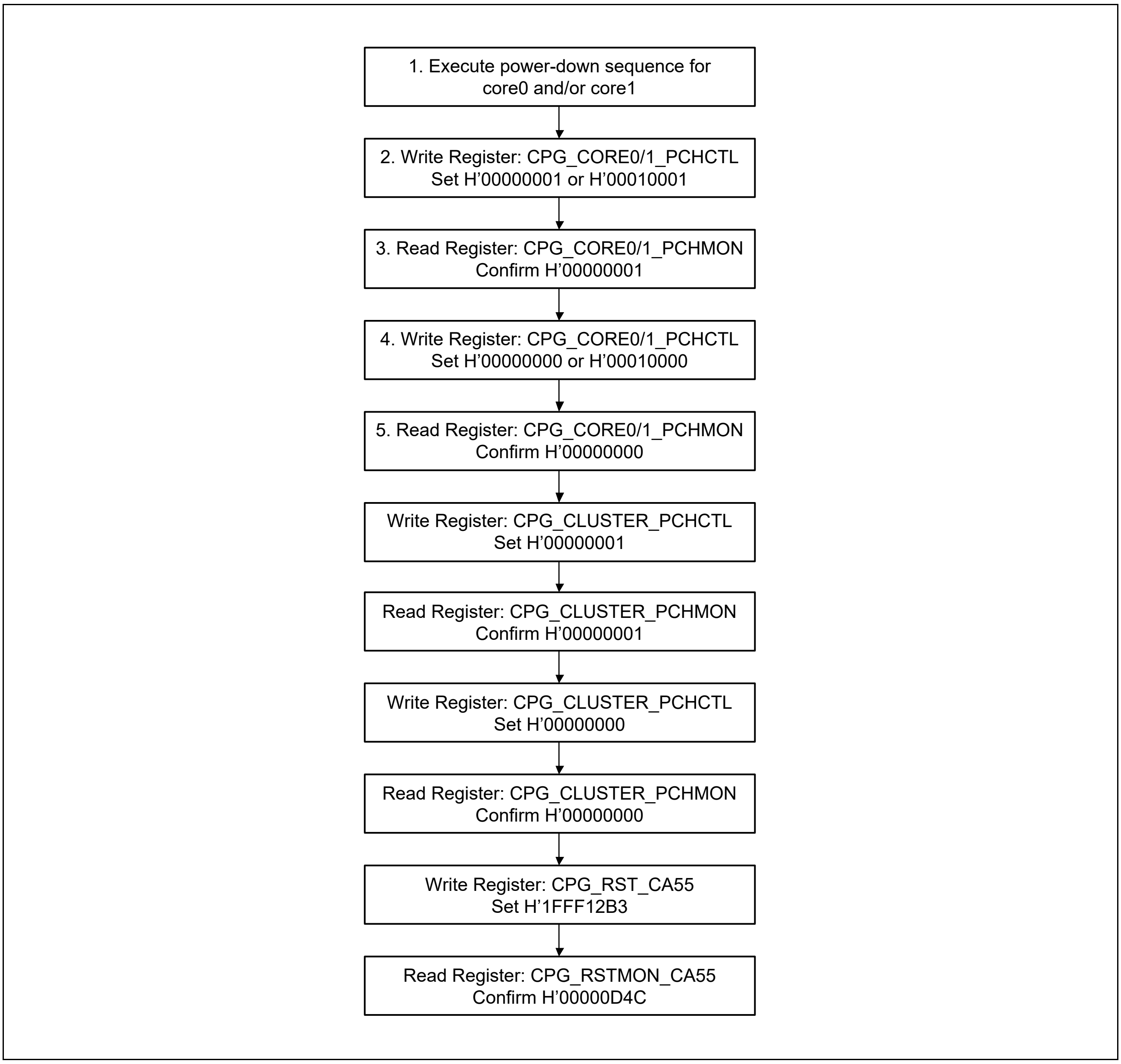
To start core 1, read “core 0” as “core 1” in the sentences below.



2.2.3 Cluster Warm Reset

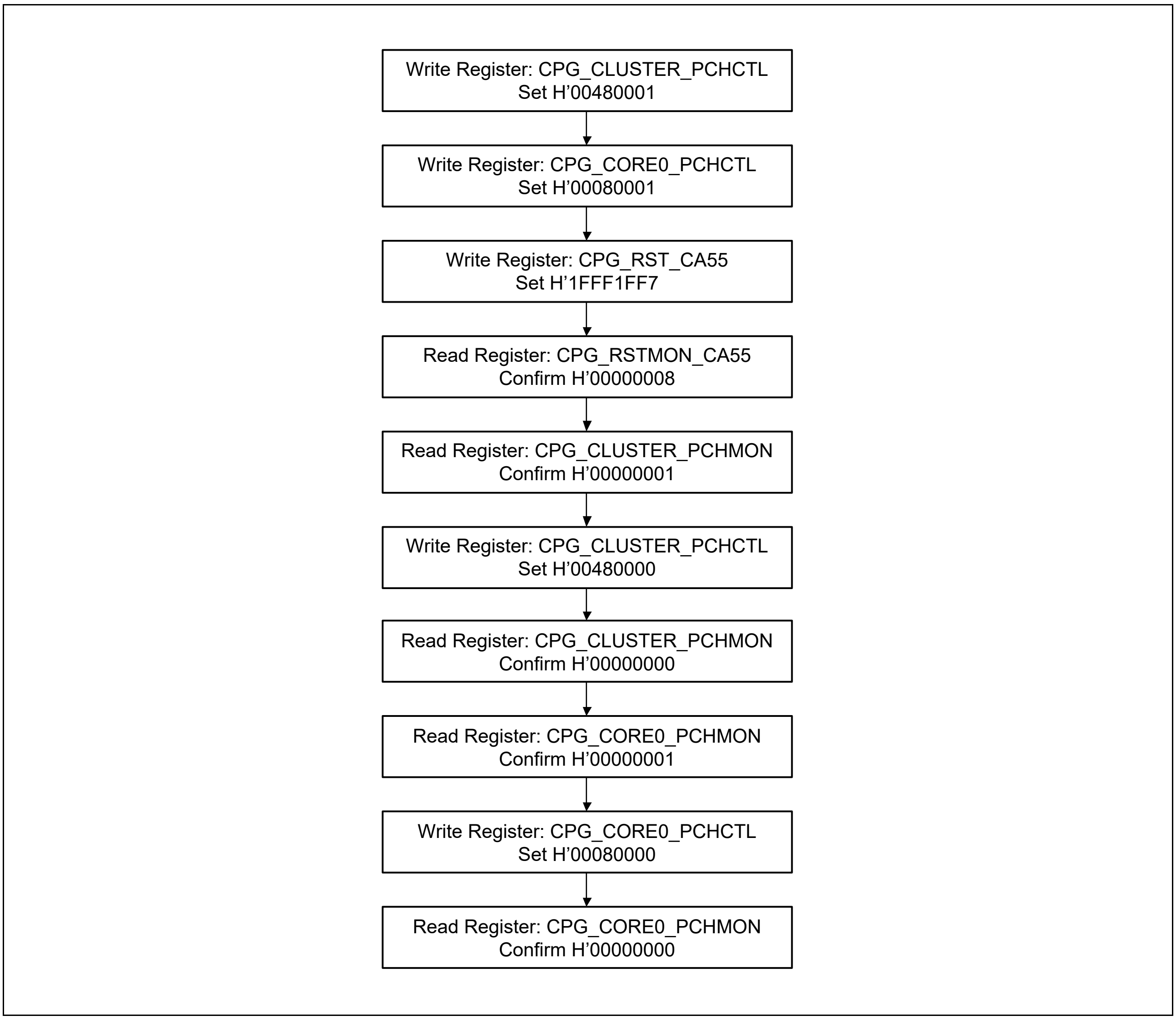
This reset requires software control by the Cortex-A55 and Cortex-M33 system CPUs.

To apply this reset for core0 and/or core1, follow the procedure below.



To release from the cluster warm reset state and start core 0, follow the procedure below.

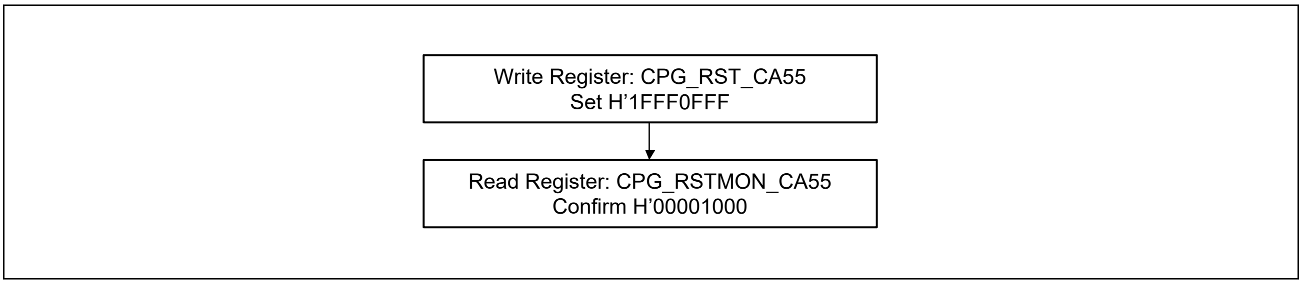
The procedure to start core 1 after having started core 0 is the same as that in the power-on reset sequence. For details, see the chapter on the clock pulse generator (CPG).



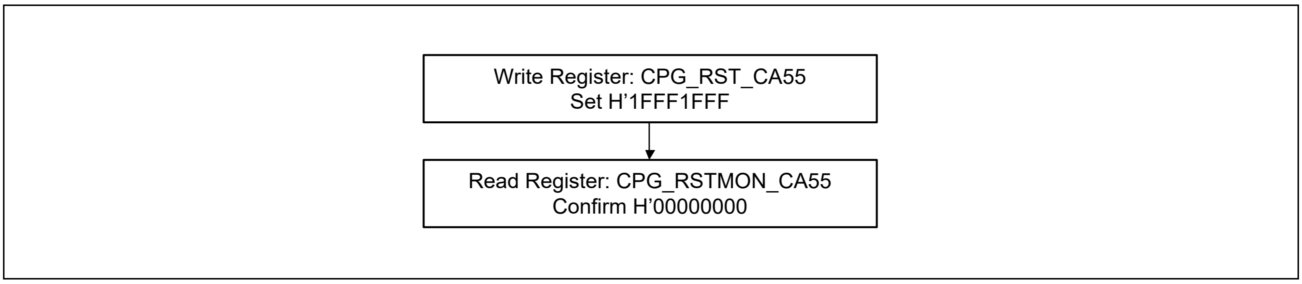
2.2.4 Debug Reset

Debug resets can only be applied under software control.

To apply a debug reset, follow the procedure below.



To release from the debug reset state, follow the procedure below.

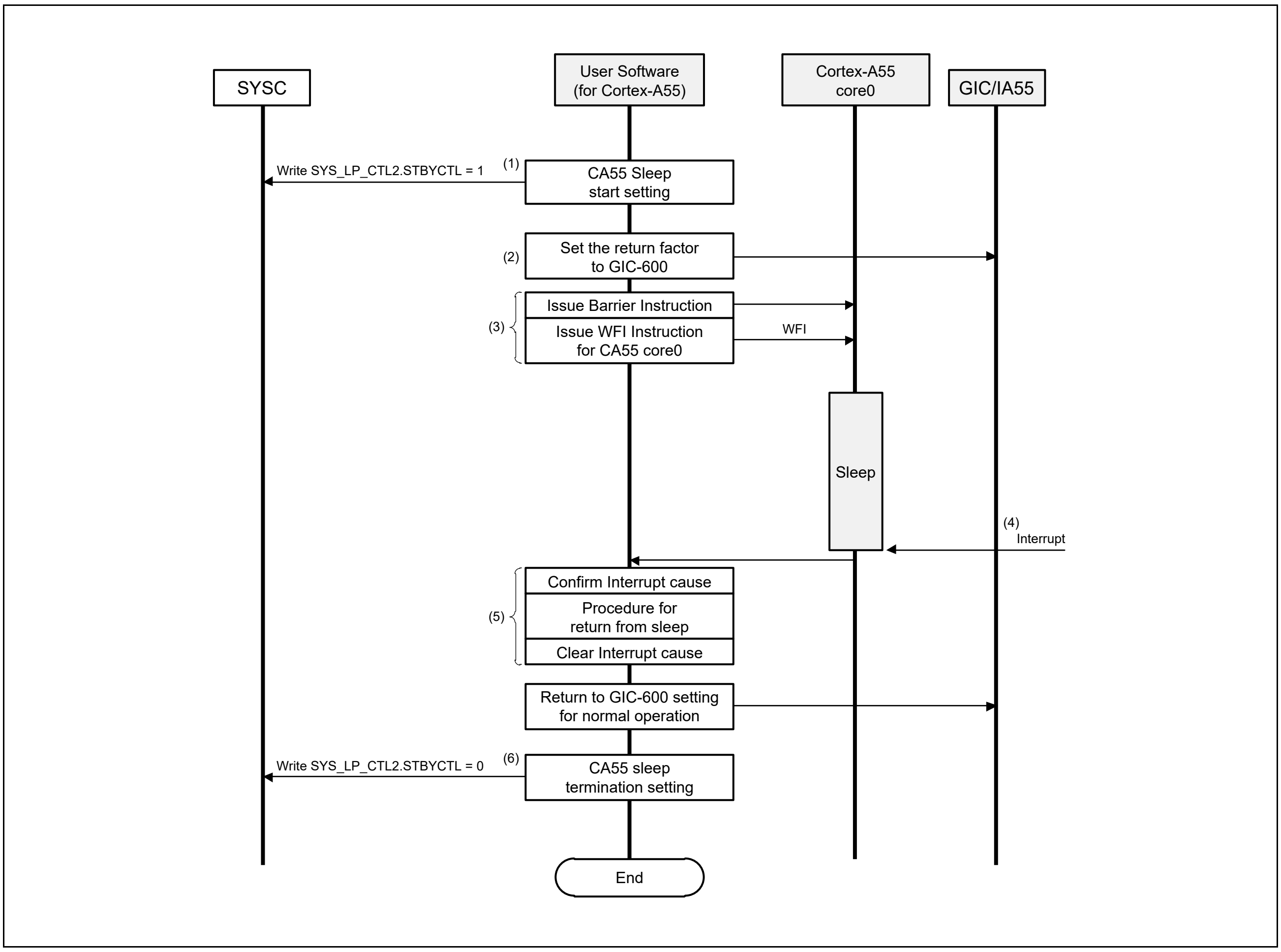


2.3 Low Power Consumption Mode

2.3.1 Cortex-A55 Sleep Mode

Cortex-A55 Sleep Mode of this LSI is a low power consumption mode obtained by having a core execute the WFI instruction. For details on the WFI instructions, see the Arm® Cortex®-A55 Core Technical Reference Manual and Arm® DynamIQ Shared Unit Technical Reference Manual.

The following procedure is the example sequence of a transition to Cortex-A55 Sleep Mode (core0).



(1) Set the STBYCTL bit of the SYS_LP_CTL2 register to 1 to start the Cortex-A55 Sleep Mode.

(2) Set the return factor to GIC. Enables only the interrupts used for return from the sleep mode and disables the interrupts used for the normal operation.

(3) Issue Barrier instruction.
Issue the WFI instruction for Cortex-A55 core0. The Cortex-A55 core0 is put into the sleep mode.

(4) Return from the sleep mode when the event occurs or the interrupt set in GIC occurs.

(5) Confirm the interrupt cause and perform the processing required for returning from the sleep mode.
And then clear the interrupt cause.
Set the end of interrupt to GIC.
Set the interrupt causes for the normal operation to GIC.

(6) Clear the STBYCTL bit of the SYS_LP_CTL2 register.

2.4 Function Reference

For details on the functions of the DynamIQ™ Shared Unit and Cortex-A55, see the documents listed below.

- Arm® DynamIQ Shared Unit Technical Reference Manual
- Arm® Cortex®-A55 Core Technical Reference Manual
- Arm® Cortex®-A55 Core Advanced SIMD and Floating-point Support Technical Reference Manual
- Arm® Cortex®-A55 Core Cryptographic Extension Technical Reference Manual

For details on clock and the controls, see the chapter on the clock pulse generator (CPG).

3. System CPU Cortex-M33

The Cortex-M33 system CPU is a core block equipped with an Arm Cortex-M33 CPU.

For details on the functions of the Cortex-M33, see the Arm® Cortex®-M33 Processor Technical Reference Manual.

Refer to **5.2.2, Cortex-M33 Address Space** in section 5 Internal BUS for CM33 address space.

3.1 Features

The Cortex-M33 processor is a highly energy-efficient processor that is intended for a microcontroller and deeply embedded applications. The processor is based on the Armv8-M architecture. For details on the functions of the Cortex-M33, see the Arm® Cortex®-M33 Processor Technical Reference Manual.

Table 3.1 Cortex-M33 Configuration

Description	Configuration
CPU	Cortex-M33 r0p4 (ARMv8-M)
Floating-point	No floating-points
DSP extension	No Armv8-M DSP extension
Security extension	Armv8-M security extension
Non-secure protected memory regions	16 regions
Secure protected memory regions	16 regions
Security attribution unit (SAU)	8 regions
Interrupts	480 interrupts
Number of bits of interrupt priority	8 bits are supported. 256 levels of priority are implemented.
Debug watchpoints and breakpoints	Full set 4 data watchpoint comparators and 8 breakpoint comparators
ITM and data watchpoint and trace (DWT) trace functionality	Complete ITM and DWT trace
Embedded trace macrocell (ETM)	ETM instruction execution trace
Micro trace buffer (MTB)	MTB is not supported.
Cross trigger interface (CTI)	CTI is included.
Wakeup interrupt controller (WIC)	WIC is included.
External coprocessor interface	Coprocessor hardware is not supported.

3.2 Address Space

For details on the address space, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual.

3.2.1 IDAU Setting

The Cortex-M33 system CPU uses the IDAU. **Table 3.2** shows the security attributes assigned by the IDAU.

Table 3.2 IDAU Security Attributes

Area No.	Security Attribute	Lower Limit of Address Range	Upper Limit of Address Range
0	Non-secure	H’0000_0000	H’0FFF_FFFF
1	Secure	H’1000_0000	H’1FFF_FFFF
2	Non-secure	H’2000_0000	H’2FFF_FFFF
3	Secure	H’3000_0000	H’3FFF_FFFF
4	Non-secure	H’4000_0000	H’4FFF_FFFF
5	Secure	H’5000_0000	H’5FFF_FFFF
6	Non-secure	H’6000_0000	H’6FFF_FFFF
7	Secure	H’7000_0000	H’7FFF_FFFF
8	Non-secure	H’8000_0000	H’8FFF_FFFF
9	Secure	H’9000_0000	H’9FFF_FFFF
10	Non-secure	H’A000_0000	H’AFFF_FFFF
11	Secure	H’B000_0000	H’BFFF_FFFF
12	Non-secure	H’C000_0000	H’CFFF_FFFF
13	Secure	H’D000_0000	H’DFFF_FFFF
14	Non-secure	H’E000_0000	H’EFFF_FFFF
15	Secure	H’F000_0000	H’FFFF_FFFF

3.3 Register Descriptions

For details on registers, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual.

For the device-specific Cortex-M33 system CPU control registers, see the section on the system controller (SYSC).

3.4 Description of Functions

For the registers and functions of Cortex-M33, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual and Arm®v8-M Architecture Reference Manual.

3.4.1 Startup Sequence

The startup sequence of the Cortex-M33 system CPU is shown in the figure below.

It is designed to execute by other processor (Cortex-A55) for the function and sequence of Cortex-M33 control here.

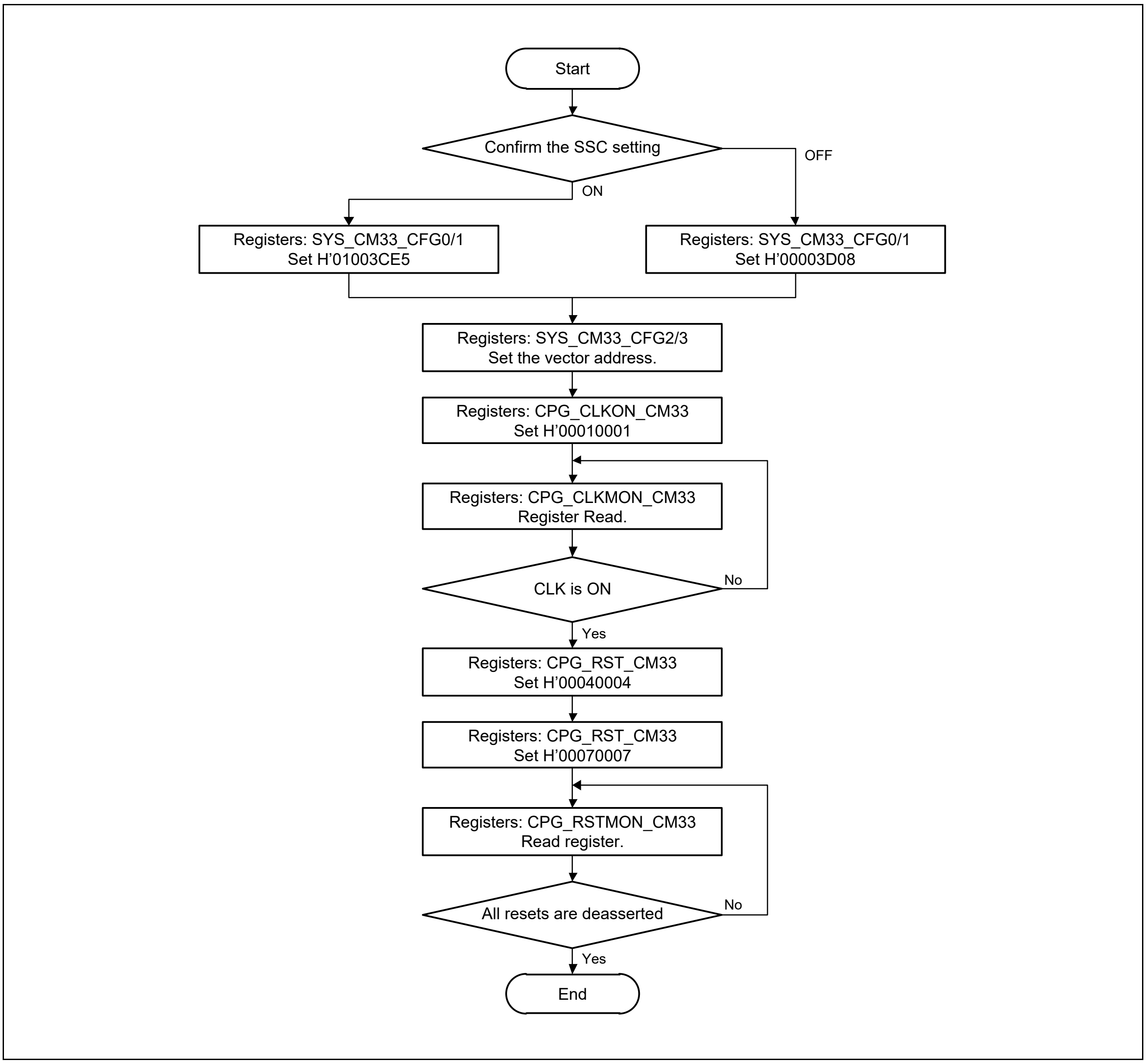


Figure 3.1 Startup Sequence

3.4.2 Control by the SYSC

In this LSI, the system controller (SYSC) is used to control the functions of the Cortex-M33 system CPU. For details of SYSC registers, see the section on the system controller (SYSC).

Table 3.3 List of Functions Controlled by the SYSC

No.	Description
1	Cortex-M33 control registers ● CM33 Config registers (SysTick timer, reset vector address): (SYS_CM33_CFGx), ● Low-power Sequence CM33 Control Register0: (SYS_LP_CM33CTL0)
2	Warm reset request Cooperation with the SYSC is necessary for executing a warm reset. For details, see **Section 3.4.3**.
3	Control of low power consumption function The Cortex-M33 system CPU supports two low power modes . For details on how to control Cortex-M33 Sleep Mode, see **Section 3.4.4**.

3.4.3 Warm Reset

When executing a warm reset, the WFI instruction is used to guarantee that no transaction is issued.

The warm reset sequence is shown below. For a details of SYSC registers, see the section on the system controller (SYSC).

1. Set the SYSRESETREQ bit in the Cortex-M33 register AIRCR to 1.
2. Confirm that the value of the SYSRESETREQ bit in the SYSC register SYS_LP_CM33CTL0 is 1.
3. Set the IM33_MASK bit in the SYSC register SYS_LP_CTL7 to 1.
4. Issue a WFI instruction.*[1]

The above procedure is used to execute a warm reset. Clearing the IM33_MASK bit and release from the warm reset state are controlled by hardware.

Note 1. If the WFE instruction is executed during a warm reset in this LSI, correct operation cannot be guaranteed.

3.4.4 Cortex-M33 Sleep Mode

In this LSI, the sleep mode for the Cortex-M33 system CPU is Cortex-M33 Sleep Mode. In this mode, some modules (SysTick timers, NVIC, etc.) of the core can operate.

The following procedure is the example sequence of a transition to Cortex-M33 Sleep Mode.

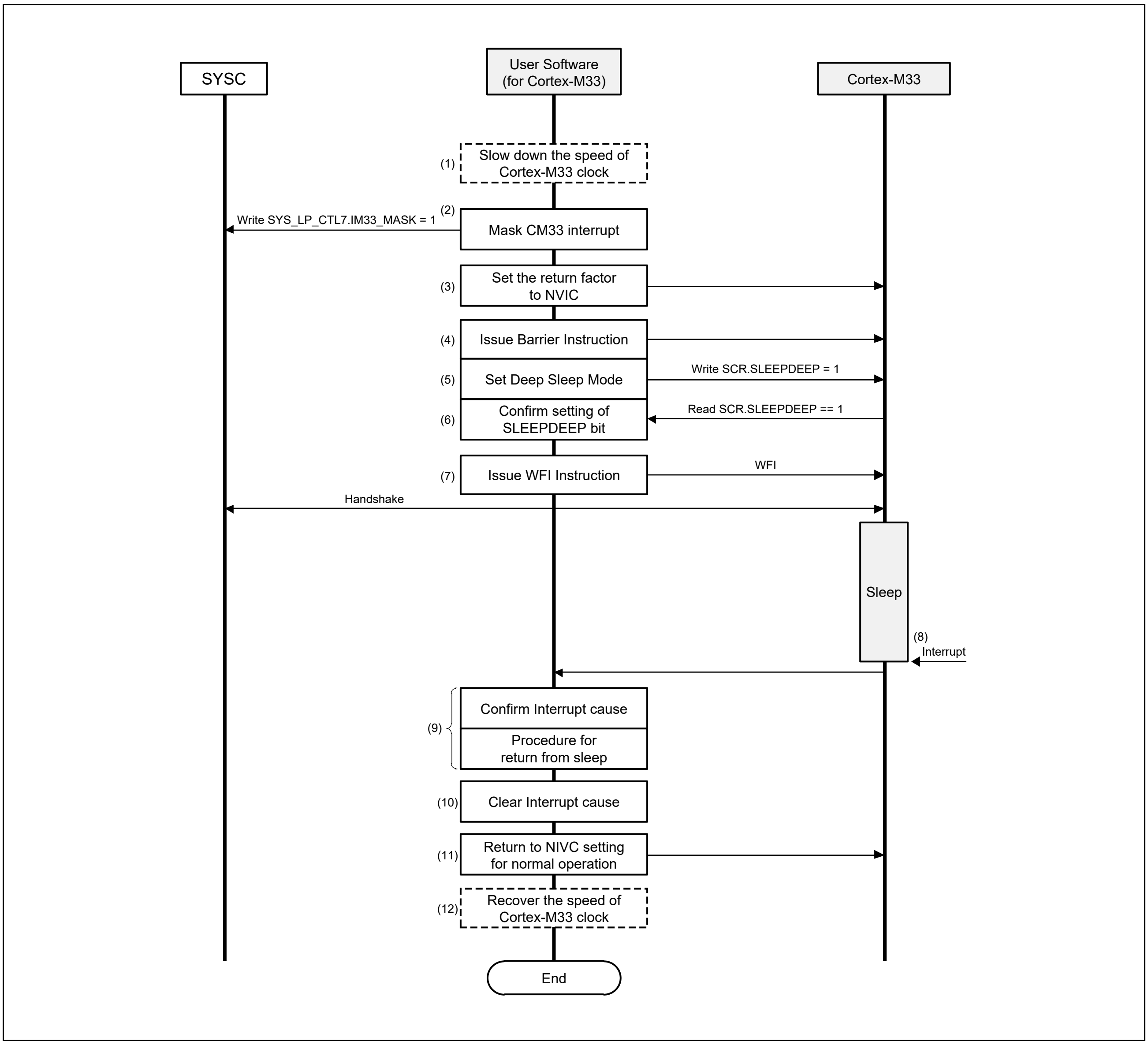


Figure 3.2 Cortex-M33 Sleep Mode procedure

(1) Slow down the speed of Cortex-M33 clock.*[1]

(2) Set the IM33_MASK bit of the SYS_LP_CTL7 register to 1.
Interrupts to IM33 are masked and interrupts from IM33 to Cortex-M33 are not accepted.
The IM33_MASK bit is automatically cleared after completing the transition to Cortex-M33 Sleep Mode.
Refer to the SYS_LP_CTL7 register in the SYSC section.

(3) Set the return factor to NVIC. Enables only the interrupts used for return from Cortex-M33 Sleep Mode and disables the interrupts used for the normal operation.*[2]

(4) Issue Barrier instruction.

(5) Set the SLEEPDEEP bit of the SCR register in Cortex-M33 to 1.*3

(6) Read SCR register to confirm whether the SLEEPDEEP bit is set to 1.

(7) Issue a WFI instruction to enter Cortex-M33 Sleep Mode.

(8) Return from Cortex-M33 Sleep Mode when the event ocurrs or the interrupt set in NVIC occurs.

(9) Confirm the interrupt cause and perform the processing required for returning from Cortex-M33 Sleep Mode.*2

(10) And then clear the interrupt cause.*2

(11) Set the interrupt causes for the normal operation to NVIC.*2

(12) Recover the speed of Cortex-M33 clock.*1

Note 1. Switching clock (I2ϕ, all Cortex-M33 clock systems) frequency is controlled by software in this LSI.
Refer to the CPG_PL3B_DDIV register in the CPG section.
Note that the step of slowing down clock (I2ϕ) is not mandatory.

Note 2. See Arm® Cortex®-M33 Processor Technical Reference Manual for NVIC.

Note 3. To make SLEEPDEEP bit 1 in the case of non-secure mode, SLEEPDEEPS bit is required to be 0. See Armv8-M Architecture Reference Manual for details.

3.4.5 SysTick Timers

The SysTick timers in the Cortex-M33 are described below.

- There is one Secure SysTick timer and one Non-secure SysTick timer.
- The count cycle of SysTick timers is common to Secure and Non-secure.
- A reference clock for counting by a SysTick timer is implemented in this LSI.
- Though the cycle of the reference clock is constant regardless of the clock(I2ϕ) cycle, the accuracy is degraded when the clock(I2ϕ) frequency is switched. Therefore, the accuracy of counting is not guaranteed when there is clock(I2ϕ) frequency switchover.
- The configuration of SysTick timers can be set in the SYSC registers SYS_CM33_CFG0 and SYS_CM33_CFG1*[1]. Before the Cortex-M33 system CPU is started up, set the following values in accordance with the PLL3 SSCG being turned on or off*[2]. SSCG clock is used for SysTick timer when turned on.

  When turned on: H’01003CE5
  When turned off: H’00003D08

The above settings are reflected in the Cortex-M33 register SYST_CALIB*[3]. The settings of the 24 lower-order bits and the 25th bit are reflected in the TENMS and SKEW bits, respectively. When the SSCG is ON, set the SKEW bit to 1. The setting of the 26th bit is reflected in the NOREF bit. Set the bit to 0 because a reference clock is implemented in this LSI.

Note 1. The register settings are reflected after release from the cold reset state.
Note 2. For details on turning on and off the SSCG, see the section on the clock pulse generator (CPG).
Note 3. For details on the SYST_CALIB register, refer to the Arm®v8-M Architecture Reference Manual.

3.4.6 Restrictions of Functions

The Cortex-M33 system CPU has some restrictions on the functions due to specification of this LSI. **Table 3.4** lists the restrictions on the functions.

Table 3.4 List of Restrictions of Functions

No.	Description
1	WFE instruction is not supported in a warm reset. In this LSI, operation of the WFE instruction is not guaranteed while a warm reset is being executed.
2	SysTick timer accuracy Though the clock(I2ϕ) frequency can be switched dynamically in this LSI, the timer accuracy cannot be guaranteed when there is a dynamic frequency switchover during the measurement period.

4. Boot Mode

4.1 Overview

This LSI has six boot modes.

Boot mode 0: Booting from eSD*[1]

Boot mode 1: Booting from 1.8-V eMMC*[2]

Boot mode 2: Booting from 3.3-V eMMC*[2]

Boot mode 3: Booting from the 1.8-V Single, Quad, or Octal*[3] serial flash memory connected to the SPI multi I/O bus space

Boot mode 4: Booting from the 3.3-V Single or Quad serial flash memory connected to the SPI multi I/O bus space

Boot mode 5: Booting from the program downloaded through the serial communications with FIFO (SCIF)

Note 1. Embedded SD (eSD) defined in the SD Specification Part 1 eSD Addendum (Version 2.10)

Note 2. eMMC supporting the boot operation mode prescribed in JEDEC STANDARD JESD84 A44 (MMCA 4.4)

Note 3. OctaFlash™ memory is supported as boot device, but HyperFlash™ memory is not supported as boot device in this LSI.

NOTES

- Boot operation is only possible with Cortex-A55 (AArch64).
- Boot operation is not possible with Cortex-M33.

Once this LSI is released from the system reset state, the clock pulse generator (CPG) executes a specified sequence, and then Cortex-A55 Core 0 is started first to boot from the device selected according to the settings through the MD_BOOT2 to MD_BOOT0 pins. The values of the MD_BOOT2 to MD_BOOT0 pins are read once the LSI is released from the system reset state. After the LSI is booted up, the user program should enable or disable the operation of Cortex-A55 Core 0, Cortex-A55 Core 1, and Cortex-M33 as required.

The CPG_PL1_DDIV register must be set when the user program starts.

Table 4.1 Selection of Boot Mode

MD_BOOT2 to MD_BOOT0			Boot Mode	Interface Module	Connected Device	Reference Section
0	0	0	Boot mode 0	SDHI0	eSD (3.3 V at startup)	**4.2.1**
0	0	1	Boot mode 1	SDHI0	1.8-V eMMC	**4.2.2**
0	1	0	Boot mode 2	SDHI0	3.3-V eMMC	**4.2.3**
0	1	1	Boot mode 3	SPIBSC	1.8-V Single, Quad, or Octal serial flash memory	**4.2.4**
1	0	0	Boot mode 4	SPIBSC	3.3-V Single or Quad serial flash memory	**4.2.5**
1	0	1	Boot mode 5	SCIF0	Downloading through SCIF	**4.2.6**
1	1	0	Reserved	—	Reserved	—
1	1	1	Reserved	—	Reserved	—

4.2 Operation

4.2.1 Boot Mode 0 (eSD)

Table 4.2 shows the interface signals used for connection with the external device in boot mode 0 (eSD). This LSI supports the embedded SD (eSD) defined in the SD Specification Part 1 eSD Addendum (Version 2.10).

Table 4.2 External Interface Signals Used in Boot Mode 0

Interface Module	Pin Name	I/O	Function	Pin Type
SDHI/eMMC (SDHI0)	SD0_CLK	Output	SD clock	Dedicated pins
	SD0_CMD	Input/output	SD command or response	
	SD0_DATA0	Input/output	SD data 0	
	SD0_DATA1	Input/output	SD data 1	
	SD0_DATA2	Input/output	SD data 2	
	SD0_DATA3	Input/output	SD data 3	
	SD0_CD	Input	SD card detection	Multiplexed pins
	SD0_WP	Input	SD write protection	

4.2.1.1 External Connections

Figure 4.1 shows the connections with the eSD device.

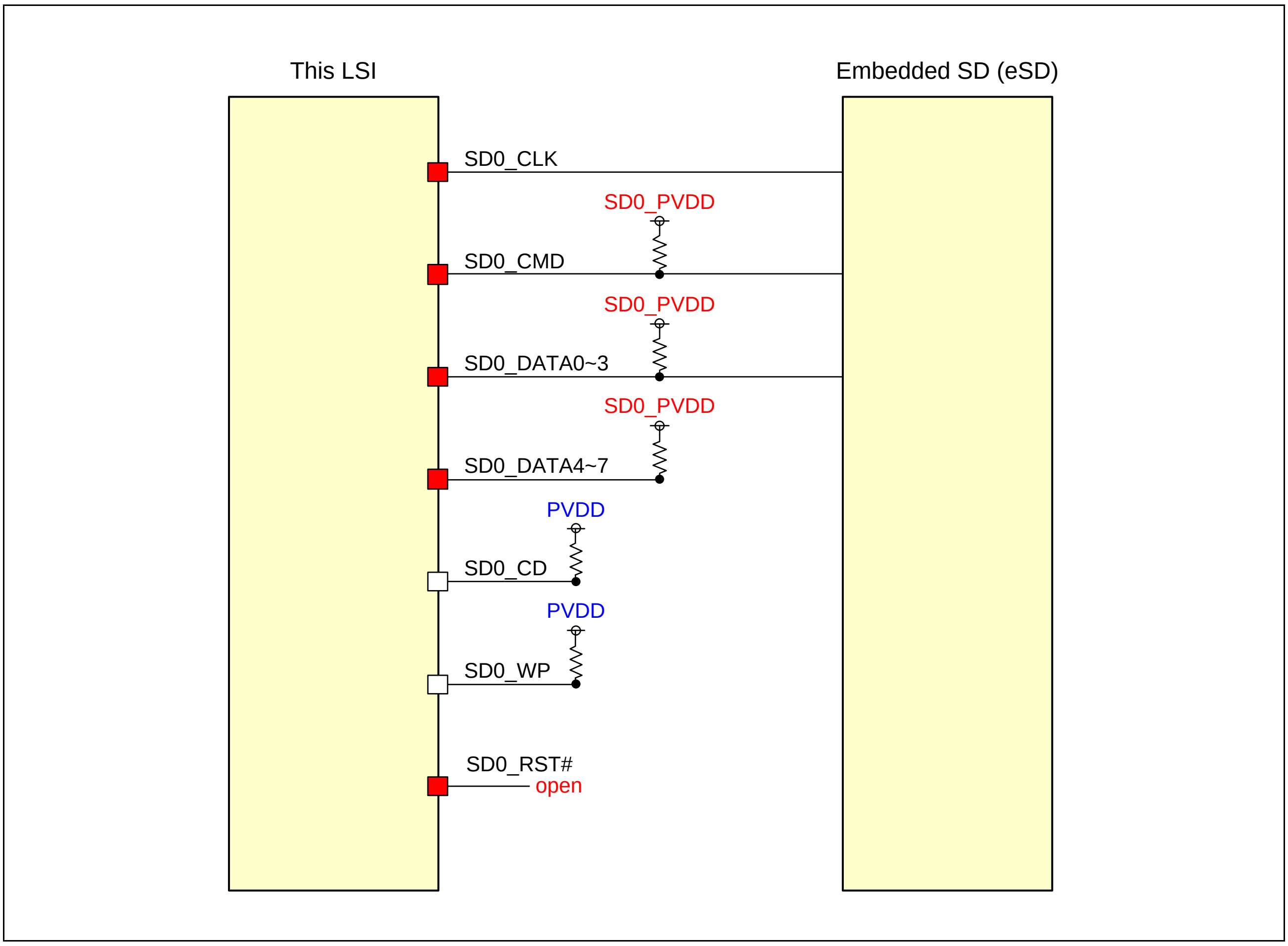


Figure 4.1 Connections in Boot Mode 0

4.2.1.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the eSD device. The operating voltage is set to 3.3 V and the width of the data bus connected to the eSD device is fixed to four bits in this mode. SDHI0 used as the interface controller in this mode is placed in the module standby mode at the startup of the LSI. Once SDHI0 is released from the module standby mode, the boot program in this LSI sets up the CPG so that the frequency of the external output clock becomes 16.6 MHz, initializes SDHI0, and then begins the booting process with the I/O buffers set to the maximum driving ability.

4.2.1.3 Operation of Booting from eSD

In boot mode 0 (booting from eSD), this LSI is booted from the loader program in the eSD device connected to channel 0 of the SD host interface (SDHI) as follows.

The boot program executes the following processing to access the eSD device.

1. Setting up the necessary peripheral modules (SDHI channel 0 and GPIO)
2. Mounting the eSD device
3. Issuing a read command to the eSD device through the SDHI to obtain the size of the loader program data from sector 1 of the selected partition

After obtaining the size of the loader program data, the SDHI issues a read command to the eSD device to transfer the loader program from sector 8 of the selected partition to the addresses H’0_0001_2000 to H’0_0002_EFFF of the on-chip RAM for the obtained data size.

Then, execution branches to the start address (H’0_0001_2000) of the loader program transferred to the on-chip RAM to execute the loader program that was stored in the eSD device.

If the boot program has failed to read data, it reads data from a reserved area. If reading has failed for all reserved areas, the boot program enters fail-safe mode (SCIF downloading mode). If the fail-safe processing has failed, execution enters an infinite loop in the on-chip ROM and the boot processing is terminated.

4.2.1.4 Allocation of the Loader Programs in the eSD Device

(1) Allocation in eSD V2.0 (Single Partition)

Figure 4.2 shows the allocation of the loader program size blocks and loader programs in the eSD device conforming to the eSD V2.0 standard. To prevent read-disturb errors, up to seven loader program size blocks and loader programs can be multiplexed and written to the eSD device.

The loader program size blocks are allocated to sectors 1 to 7. In a loader program size block, the loader program size is stored in the first four bytes and the signature H'AA55 is stored in the last two bytes. **Figure 4.3** shows the structure of the loader program size data. Be sure to store the loader program size in 4-byte little endian.

The following describes the operation of transferring the loader program.

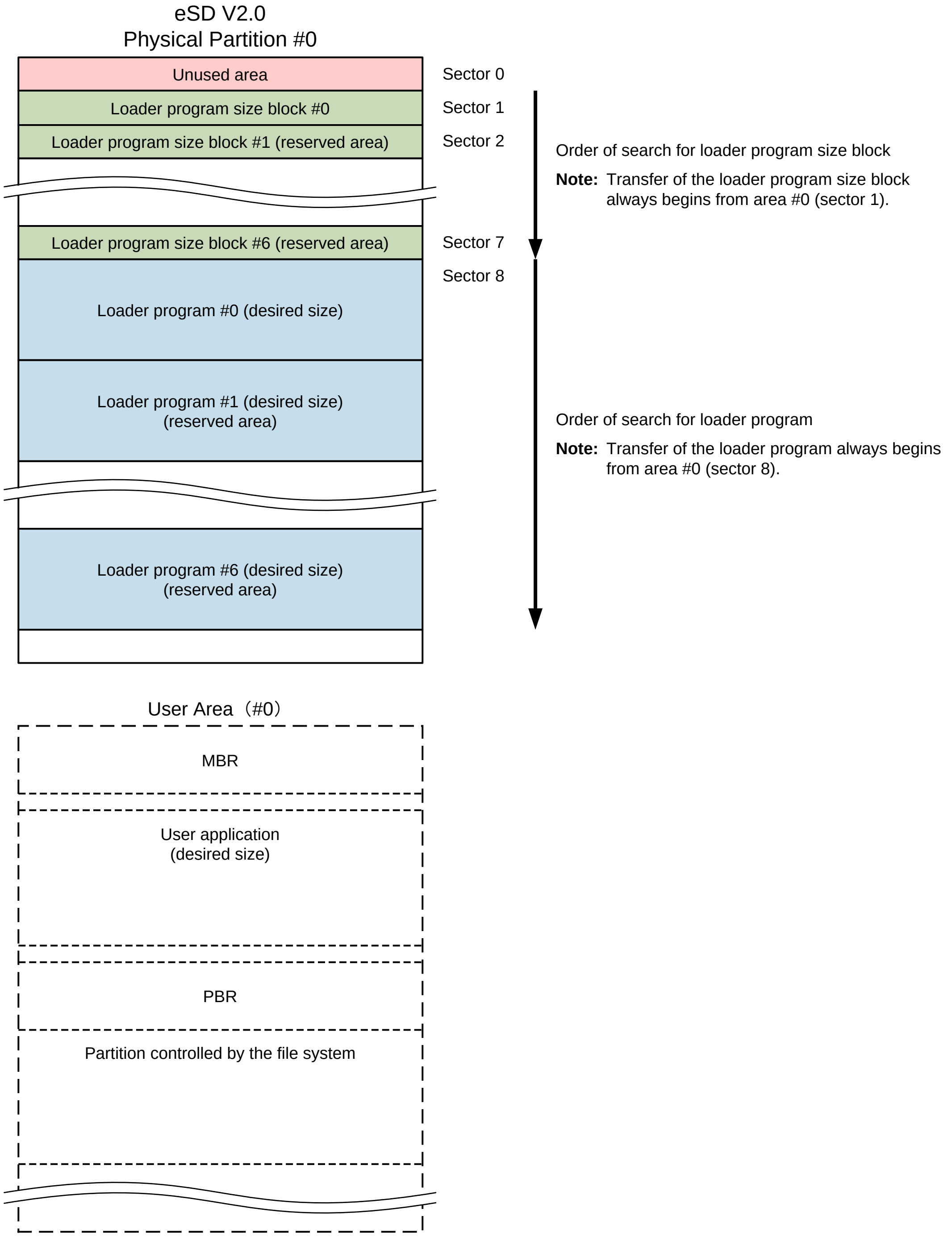
(1) The loader program size block is transferred from sector 1 in physical partition #0 of the eSD device to the on-chip RAM.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the transfer of the loader program size block in a single area, another loader program size block is transferred from the next multiplexed area.

(2) The signature in the transferred loader program size block is checked. When the signature matches the expected value (H'AA55), the size of the loader program data is obtained from the loader program size block and the processing for transferring the loader program is executed.
If the signature does not match the expected value, another loader program size block is transferred from the next multiplexed area.

(3) The transfer of the loader program begins from sector 8 in physical partition #0 of the eSD device. The loader program is transferred to the on-chip RAM for the obtained loader program size.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the loader program transfer in a single area, another loader program is transferred from the next multiplexed area.

NOTE

Even when the loader size block is transferred from a multiplexed (reserved) area, the transfer of the loader program always begins from the area for loader program #0.



Note 1. Write the same data to loader program size block areas #0 to #6.

Note 2. Write the same data to loader program areas #0 to #6.

Note 3. The start sector location of the loader program is calculated as follows.
8 + obtained size (number of sectors) × n

Figure 4.2 Allocation of Loader Programs in the eSD Device Conforming to the eSD V2.0 Standard

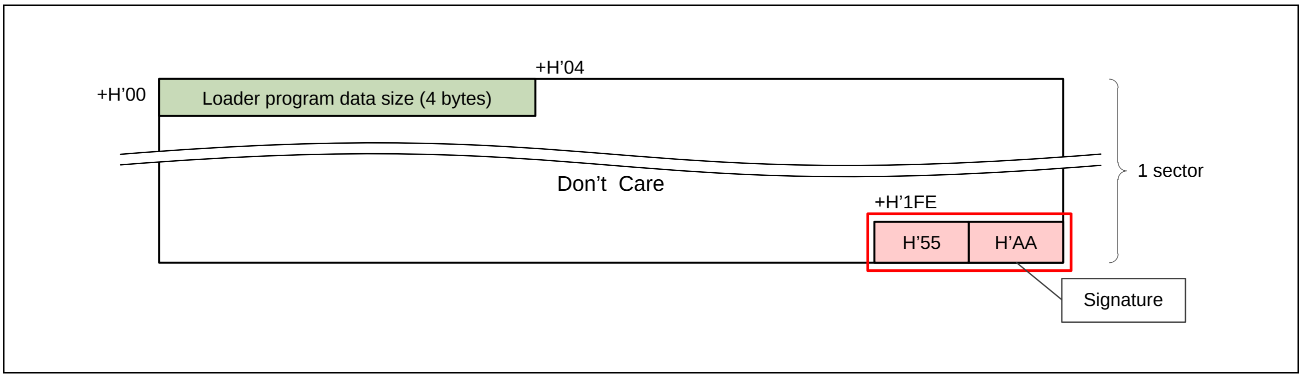


Figure 4.3 Structure of the Loader Program Size Block

NOTE

Store the loader program data size in 4-byte little endian.

(2) Allocation in eSD V2.1 (Multi Partitions)

Figure 4.4 shows the allocation of the loader program size blocks and loader programs in the eSD device conforming to the eSD V2.1 standard. To prevent read-disturb errors, up to seven loader program size blocks and loader programs can be multiplexed and written to the eSD device

The specifications and allocation of loader program size blocks are the same as those in eSD V2.0. For the structure of the loader program size block, see **Figure 4.3**.

The following describes the operation of transferring the loader program.

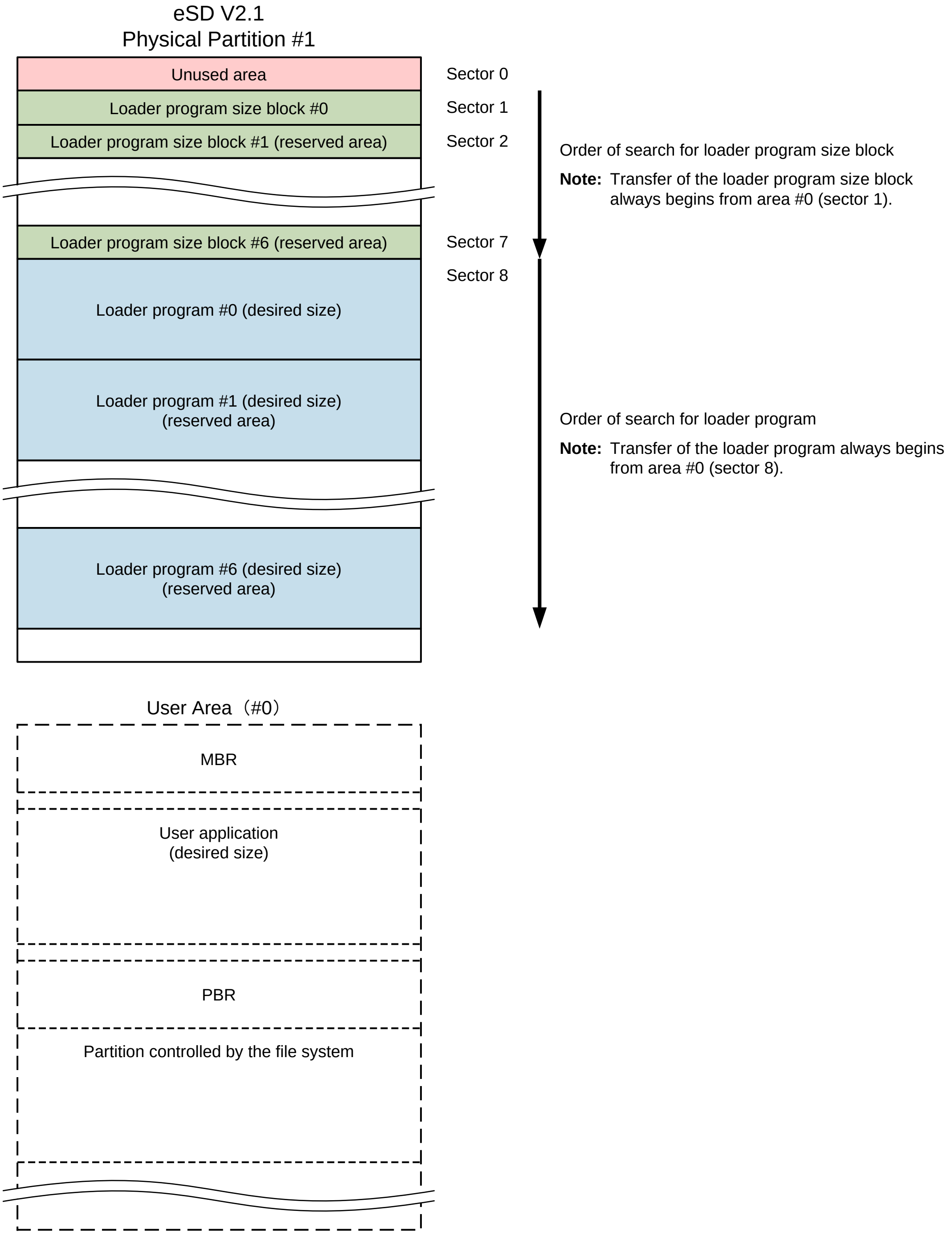
(1) The loader program size block is transferred from sector 1 in physical partition #1 of the eSD device to the on-chip RAM.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the transfer of the loader program size block in a single area, another loader program size block is transferred from the next multiplexed area.

(2) The signature in the transferred loader program size block is checked. When the signature matches the expected value (H'AA55), the size of the loader program data is obtained from the loader program size block and the processing for transferring the loader program is executed. If the signature does not match the expected value, another loader program size block is transferred from the next multiplexed area.

(3) The transfer of the loader program begins from sector 8 in physical partition #1 of the eSD device. The loader program is transferred to the on-chip RAM for the obtained loader program size.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the loader program transfer in a single area, another loader program is transferred from the next multiplexed area.

NOTE

Even when the loader size block is transferred from a multiplexed (reserved) area, the transfer of the loader program always begins from the area for loader program #0.



Note 1. Write the same data to loader program size block areas #0 to #6.

Note 2. Write the same data to loader program areas #0 to #6.

Note 3. The start sector location of the loader program is calculated as follows.
8 + obtained size (number of sectors) × n

Figure 4.4 Allocation of Loader Programs in the eSD Device Conforming to the eSD V2.1 Standard

(3) How to Distinguish between eSD V2.0 (Single Partition) and V2.1 (Multi Partitions)

This boot program first assumes that an eSD device supporting multi partitions is connected and begins loader program transfer from physical partition #1. If an error regarding a command for multi partitions shown in the following table occurs, the boot program assumes that an eSD device with a single partition is connected and transfers the loader program from physical partition #0.

Table 4.3 Errors Regarding Commands for Multi Partitions

No.	Issued Command	Command Function	Error	Remarks
1	CMD43 (SELECT_PARTITION)	Selects physical partition #1.	Multi partitions are not supported.	
2	CMD45 (QUERY_PARTITION)	Obtains the physical partition ID.	The specified partition does not exist.	

4.2.1.5 Note

The shared bus and 8-bit SD bus of the embedded SDIO cannot be used in this mode.

4.2.2 Boot Mode 1 (1.8-V eMMC)

Table 4.4 shows the interface signals used for connection with the external device in boot mode 1 (eMMC). This LSI supports the eMMC that operates in the boot operation mode prescribed in JEDEC STANDARD JESD84 A44 (MMCA 4.4).

Table 4.4 External Interface Signals Used in Boot Mode 1

Interface Module	Pin Name	I/O	Function	Pin Type
SDHI/eMMC (SDHI0)	SD0_CLK	Output	eMMC clock	Dedicated pins (1.8 V)
	SD0_CMD	Input/output	eMMC command or response	
	SD0_DATA0	Input/output	eMMC data 0	
	SD0_DATA1	Input/output	eMMC data 1	
	SD0_DATA2	Input/output	eMMC data 2	
	SD0_DATA3	Input/output	eMMC data 3	
	SD0_DATA4	Input/output	eMMC data 4	
	SD0_DATA5	Input/output	eMMC data 5	
	SD0_DATA6	Input/output	eMMC data 6	
	SD0_DATA7	Input/output	eMMC data 7	
	SD0_CD	Input	SD card detection	
	SD0_WP	Input	SD write protection	
	SD0_RST#	Output	eMMC reset	

4.2.2.1 External Connections

Figure 4.5 shows the connections with the eMMC device. The boot program in this LSI does not monitor the state of the SD0_CD and SD0_WP pins.

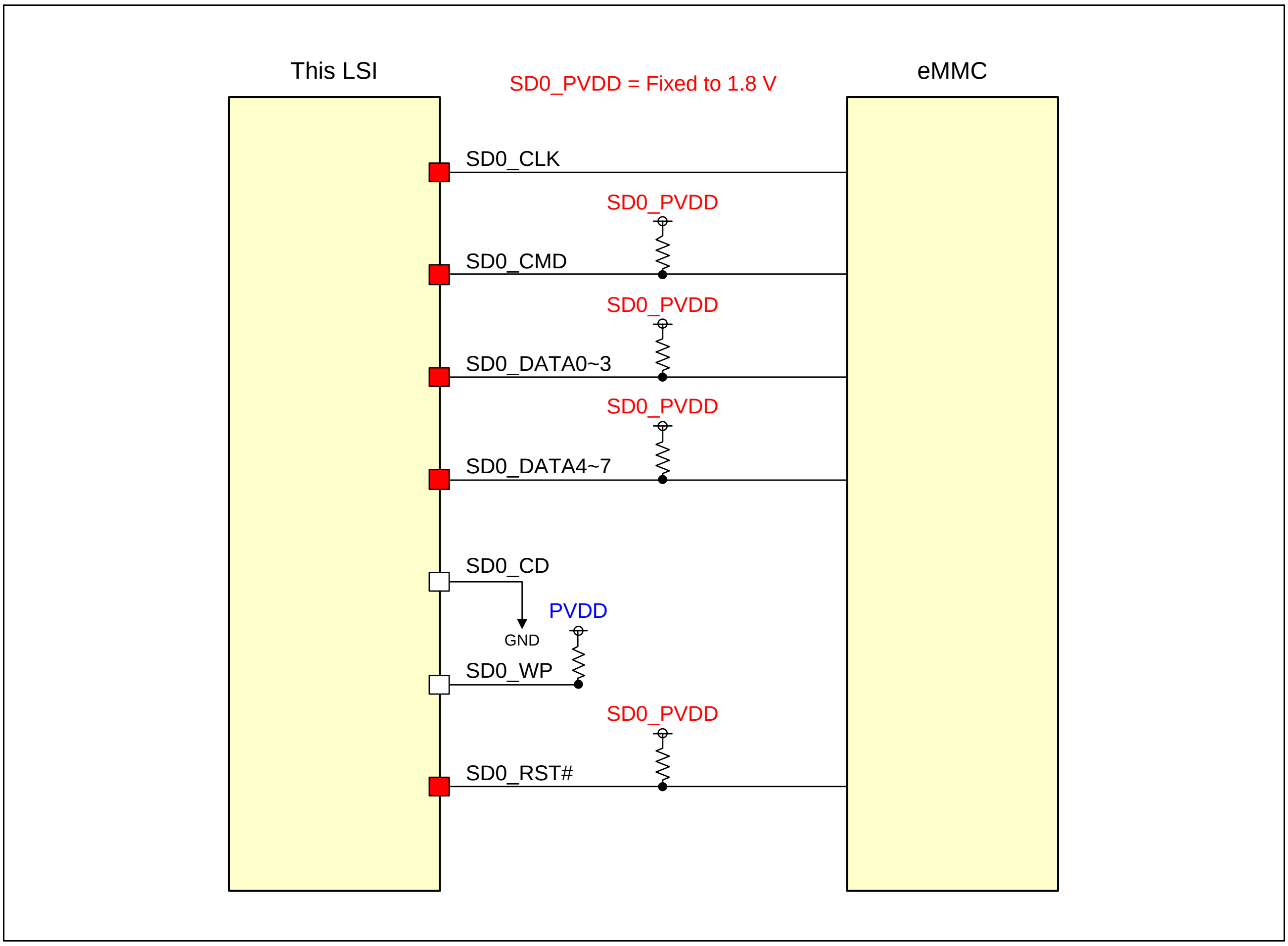


Figure 4.5 Connections in Boot Mode 1

4.2.2.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the eMMC device. The operating voltage is set to 1.8 V and the width of the data bus connected to the eMMC device is fixed to eight bits in this mode. SDHI0 used as the interface controller in this mode is placed in the module standby mode at the startup of the LSI. Once SDHI0 is released from the module standby mode, the boot program in this LSI sets up the CPG so that the frequency of the external output clock becomes 16.6 MHz, initializes SDHI0, and then begins the booting process.

4.2.2.3 Operation of Booting from 1.8-V eMMC

In boot mode 1 (booting from 1.8-V eMMC), this LSI is booted from the loader program in the 1.8-V eMMC device connected to channel 0 of the SD host interface (SDHI) as follows.

The boot program executes the following processing to access the eMMC device.

1. Setting up the necessary peripheral modules (SDHI channel 0 and OSTM channel 0)
2. Starting the alternative boot operation mode of the eMMC device
   Read access begins from sector 0 of the partition selected by the [179] field (PARTITION_CONFIG) of the extended CSD (EXT_CSD) register in the eMMC device. For details, refer to **Section 4.2.2.5** and **Section 4.2.2.5(1)**.
3. Dummy reading from sector 0 and obtaining the size of the loader program data from sector 1

After the size of the loader program data is obtained, the loader program is transferred from sector 2 to the addresses H’0_0001_2000 to H’0_0002_EFFF of the on-chip RAM for the obtained data size.

Then, execution branches to the start address (H’0_0001_2000) of the loader program transferred to the on-chip RAM to execute the loader program that was stored in the eMMC device.

If the boot program has failed to read the loader program, it enters fail-safe mode (SCIF downloading mode). If the fail-safe processing has failed, execution enters an infinite loop in the on-chip ROM and the boot processing is terminated.

Note that the data bus width is set to eight bits for booting from the eMMC device.

4.2.2.4 Allocation of the Loader Program in the eMMC Device

Figure 4.6 shows the allocation of the loader program in the eMMC device conforming to the MMCA 4.4 standard.

In the loader program size block, the loader program size is stored in the first four bytes and the signature H’AA55 is stored in the last two bytes. **Figure 4.7** shows the structure of the loader program size data. Be sure to store the loader program size in 4-byte little endian.

The transfer of the loader program begins from the first address (sector 0) of the boot partition. A dummy read from sector 0 is done, and then the size of the loader program data is obtained from the data read from sector 1. The loader program is transferred from sector 2 to the addresses H’0_0001_2000 to H’0_0002_EFFF of the on-chip RAM for the obtained loader program size.

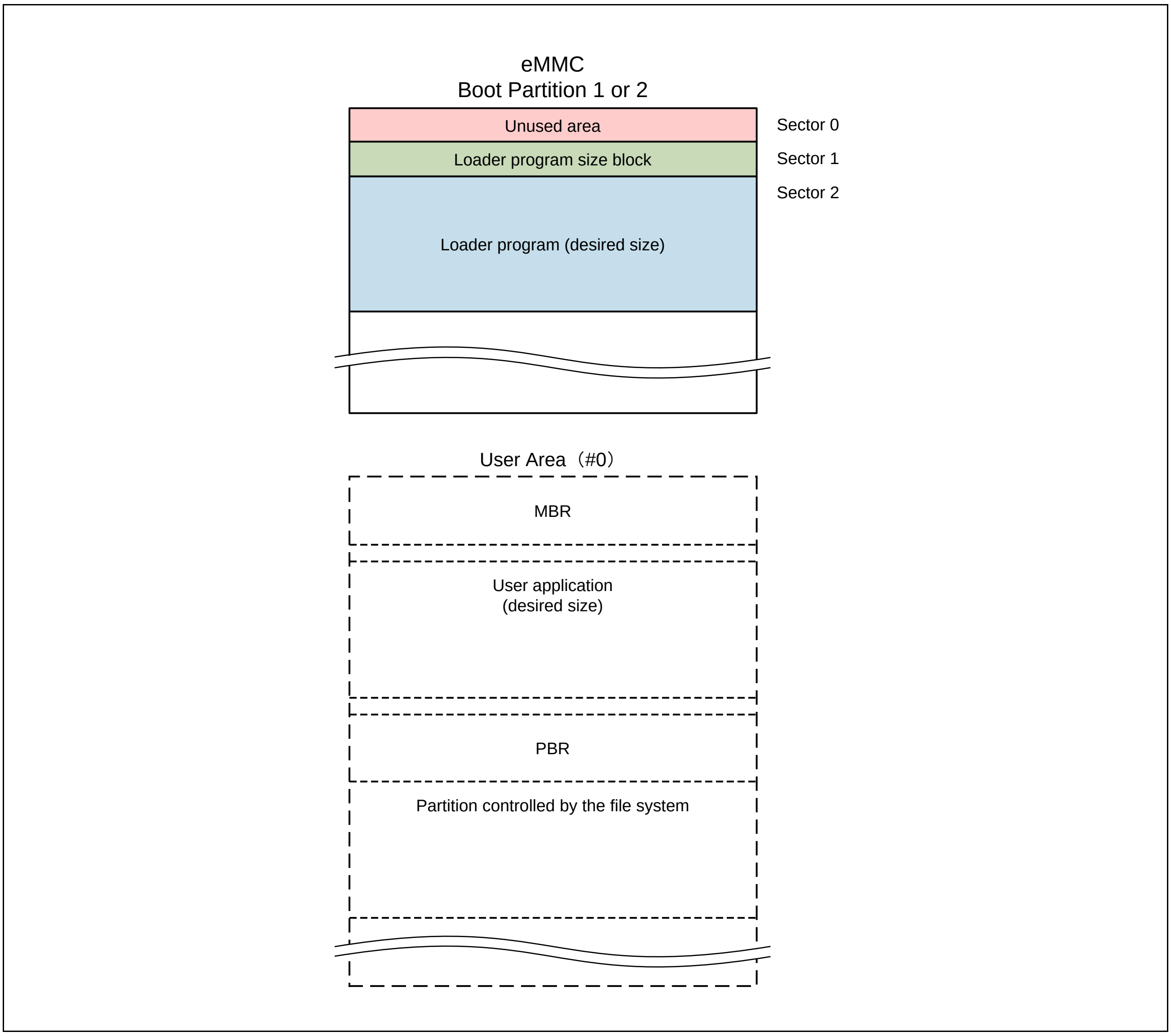


Figure 4.6 Allocation of Loader Program in the eMMC Device

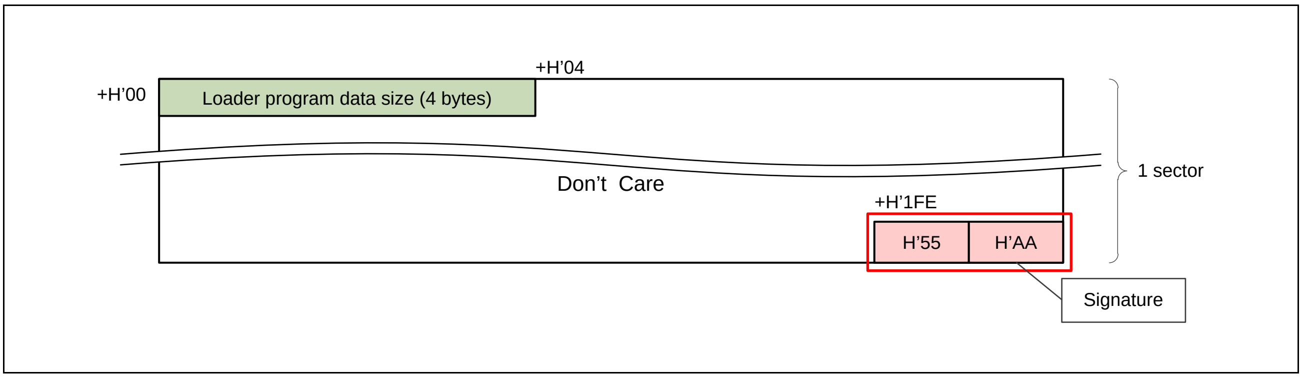


Figure 4.7 Structure of the Loader Program Size Block

In the eMMC device, the loader program is read from the partition specified by the BOOT_PARTITION_ENABLE bits in the PARTITION_CONFIG (ECSD[179]) field of the extended CSD register. Write the loader program to the partition selected by this register. **Table 4.5** shows the specifications of the extended CSD register.

Table 4.5 Specifications of the Extended CSD Register

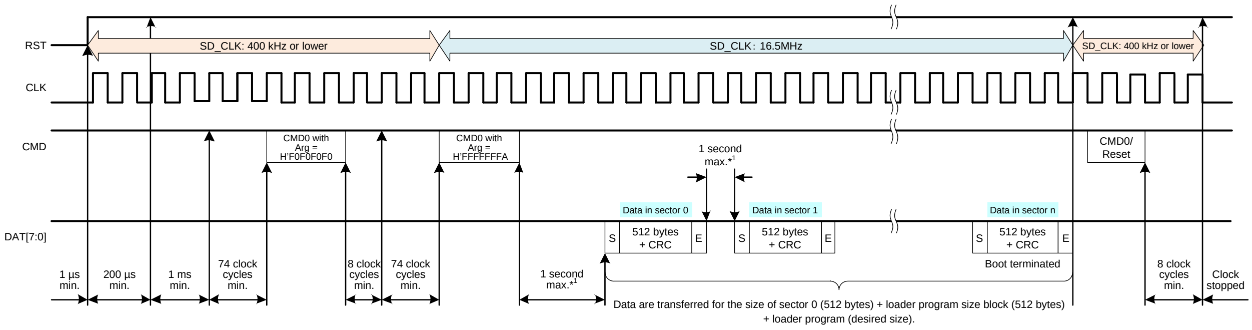
No.	Field Name	Bit Name	Setting	Remarks
1	PARTITION_CONFIG (ECSD[179])	BOOT_PARTITION_ENABLE bits [5:3]	H'0: Device not boot enabled (default) H'1: Boot partition 1 enabled for boot H'2: Boot partition 2 enabled for boot H'7: User area enabled for boot	

Allocate partitions as follows.

- Store the loader program in boot partition 1.
- Store the application program in boot partition 2.
- Allocate the user area to a partition controlled by the file system.

4.2.2.5 Alternative Boot Operation

Figure 4.8 shows the alternative boot operation to read the loader program. This boot program sets up the registers in SDHI channel 0 so that the SDHI operates with an 8-bit bus width when booting from the eMMC device. On the eMMC device side, the data bus width in the alternative boot operation is determined by the setting of the BOOT_BUS_WIDTH[177] field in the extended CSD register. To boot up from the eMMC device correctly, set up the extended CSD register in the eMMC device as shown in **Table 4.6** when writing a program to the eMMC device.



- This CPU accesses the eMMC device with a data bus width of 8 bits in the alternative boot operation. The data bus width of the eMMC device can be specified in the BOOT_BUS_WIDTH[177] field of the extended CSD register. When writing a loader program to the eMMC device, specify the 8-bit bus width in this field.
- The eMMC device sequentially outputs data from sector 0 of the partition selected in the PARTITION_CONFIG[179] field of the extended CSD register in the alternative boot operation.
- This CPU does not detect Boot Ack.
  The output of Boot Ack from the eMMC device can be specified in the PARTITION_CONFIG[179] field. Be sure to stop the output of Boot Ack by using this field.

Note 1. For a maximum of one second, the BRE, ERR3, and ERR1 bits of the SD_INFO2 register are polled to check whether data have been received, a timeout has occurred, and a CRC error has occurred, respectively.
<Normal operation>
SD_INFO2.BRE = 1
Data are transferred from SD_BUF to the on-chip RAM.
<Error>
CRC error: SD_INFO2.ERR1 = 1
The loader program is read from a reserved area.
Timeout error: SD_INFO2.ERR3 = 1
Execution of the boot program for eMMC is terminated.

Figure 4.8 Alternative Boot Operation to Read the Loader Program

Table 4.6 Setting of the Data Bus Width in Alternative Boot Operation

Field in Extended CSD Register of eMMC Device	Setting
BOOT_BUS_WIDTH [177]	Bits 1 and 0: BOOT_BUS_WIDTH = 10b (The bus width is set to 8 bits in alternative boot operation.)

(1) Partitions

An eMMC device has multiple partitions (**Figure 4.9**). The partition used for booting can be specified in the PARTITION_CONFIG[179] field of the extended CSD (EXT_CSD) register. To boot up correctly, specify the boot partition in the extended CSD (EXT_CSD) register of the eMMC device as shown in **Table 4.7** when writing a program to the eMMC device.

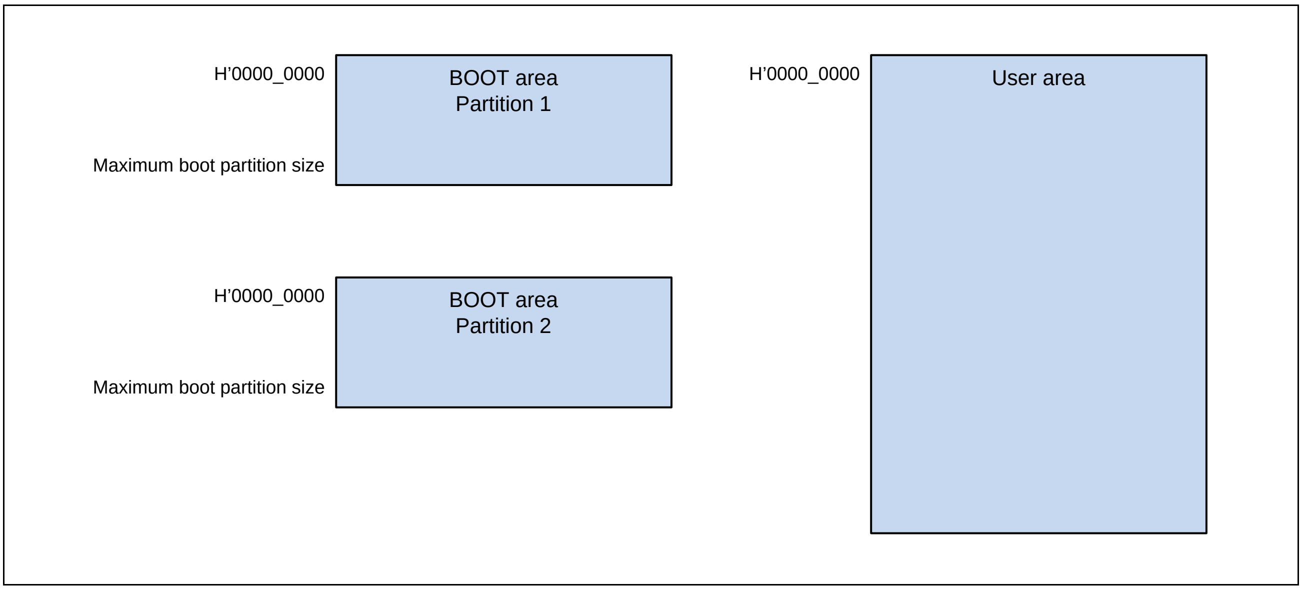


Figure 4.9 Partitions in the eMMC Device

Table 4.7 Boot Partition Setting in the Alternative Boot Operation

Field in Extended CSD Register of eMMC Device	Setting
PARTITION_CONFIG [179]	• Set bit 6 (BOOT_ACK) to 0b (the eMMC device does not output a boot acknowledge in the alternative boot operation). • Specify the boot partition in bits 5 to 3 (BOOT_PARTITION_ENABLE). 000b: Booting is disabled. 001b: Booting from sector 0 in boot partition 1 in the boot operation. 010b: Booting from sector 0 in boot partition 2 in the boot operation. 011b to 110b: Reserved 111b: Booting from sector 0 in the user area in the boot operation.

4.2.2.6 Notes

- This mode does not support booting from an MMC card.
- The width of the data bus is fixed to eight bits; 1-bit or 4-bit data bus cannot be used in this mode.

4.2.3 Boot Mode 2 (3.3-V eMMC)

Table 4.8 shows the interface signals used for connection with the external device in boot mode 2 (eMMC). This LSI supports the eMMC that operates in the boot operation mode prescribed in JEDEC STANDARD JESD84 A44 (MMCA 4.4).

Table 4.8 External Interface Signals Used in Boot Mode 2

Interface Module	Pin Name	I/O	Function	Pin Type
SDHI/eMMC (SDHI0)	SD0_CLK	Output	eMMC clock	Dedicated pins (3.3 V)
	SD0_CMD	Input/output	eMMC command or response	
	SD0_DATA0	Input/output	eMMC data 0	
	SD0_DATA1	Input/output	eMMC data 1	
	SD0_DATA2	Input/output	eMMC data 2	
	SD0_DATA3	Input/output	eMMC data 3	
	SD0_DATA4	Input/output	eMMC data 4	
	SD0_DATA5	Input/output	eMMC data 5	
	SD0_DATA6	Input/output	eMMC data 6	
	SD0_DATA7	Input/output	eMMC data 7	
	SD0_CD	Input	SD card detection	
	SD0_WP	Input	SD write protect	
	SD0_RST#	Output	eMMC reset	

4.2.3.1 External Connections

Figure 4.10 shows the connections with the eMMC device. The boot program in this LSI does not monitor the state of the SD0_CD and SD0_WP pins.

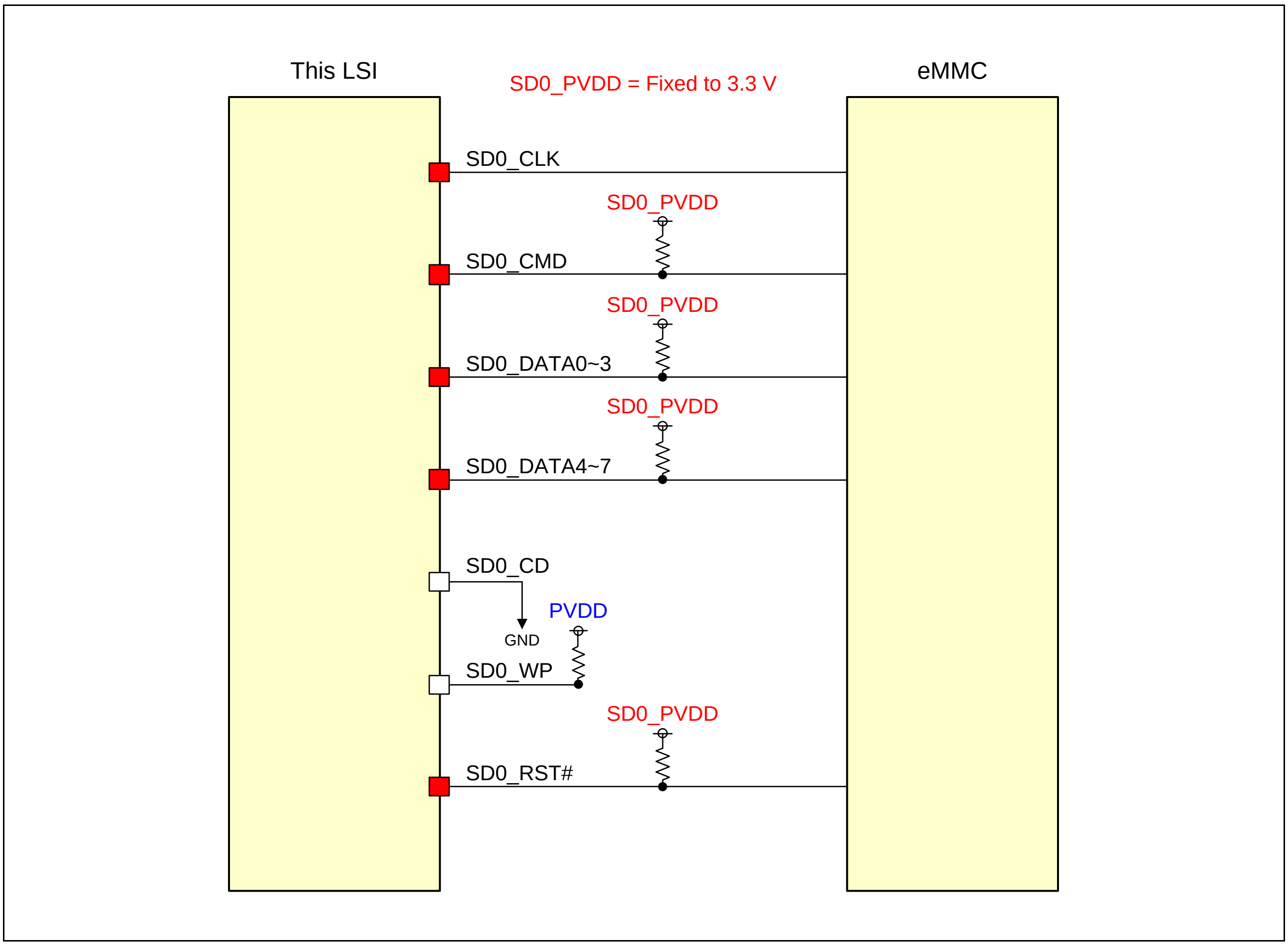


Figure 4.10 Connections in Boot Mode 2

4.2.3.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the eMMC device in the same way as boot mode 1. This mode only differs from boot mode 1 in that the external device operates at 3.3 V. The operating voltage is set to 3.3 V in this mode.

4.2.3.3 Operation of Booting from 3.3-V eMMC

Boot mode 2 (booting from 3.3-V eMMC) only differs from boot mode 1 (booting from 1.8-V eMMC) in the setting of supplied voltage as shown in **Table 4.9**.

Table 4.9 Resources Used by the Boot Program for 3.3-V eMMC (Difference from 1.8-V eMMC)

No.	Register Name	Address	Initial Value	Value after Booting	Remarks
1	GPIO_SD_CH0	H'0_1103_3000	H'0000_0000	H'0000_0000	SD0_PVDD: 0 = 3.3 V In boot mode 1 or 2, this is automatically set by hardware. The user does not need to set this register.

4.2.3.4 Notes

- This mode does not support booting from an MMC card.
- The width of the data bus is fixed to eight bits; 1-bit or 4-bit data bus cannot be used in this mode.

4.2.4 Boot Mode 3 (1.8-V Single, Quad, or Octal Serial Flash Memory)

Table 4.10 shows the interface signals used for connection with the external device in boot mode 3 (1.8-V serial flash memory).

Table 4.10 External Interface Signals Used in Boot Mode 3

Interface Module	Pin Name	I/O	Function	Pin Type
QSPI0 and QSPI1	QSPI0_SPCLK	Input/output	QSPI clock	Dedicated pins (1.8 V)
	QSPI0_IO0	Input/output	QSPI data 0	
	QSPI0_IO1	Input/output	QSPI data 1	
	QSPI0_IO2	Input/output	QSPI data 2	
	QSPI0_IO3	Input/output	QSPI data 3	
	QSPI0_SSL	Output	Slave select	
	QSPI1_SPCLK	Output	QSPI clock	
	QSPI1_IO0	Input/output	QSPI data 4	
	QSPI1_IO1	Input/output	QSPI data 5	
	QSPI1_IO2	Input/output	QSPI data 6	
	QSPI1_IO3	Input/output	QSPI data 7	
	QSPI1_SSL	Input/output	OCTAL DQS input	
	QSPI_RESET#	Output	QSPI reset	
	QSPI_WP#	Output	QSPI write protect	
	QSPI_INT#	Input	QSPI interrupt	

Note: The Octal flash memory does not have the pins shaded in gray.

4.2.4.1 External Connections

Figure 4.11 to **Figure 4.13** show the connections with the Single SPI memory, Quad SPI memory, and Octal flash memory, respectively.

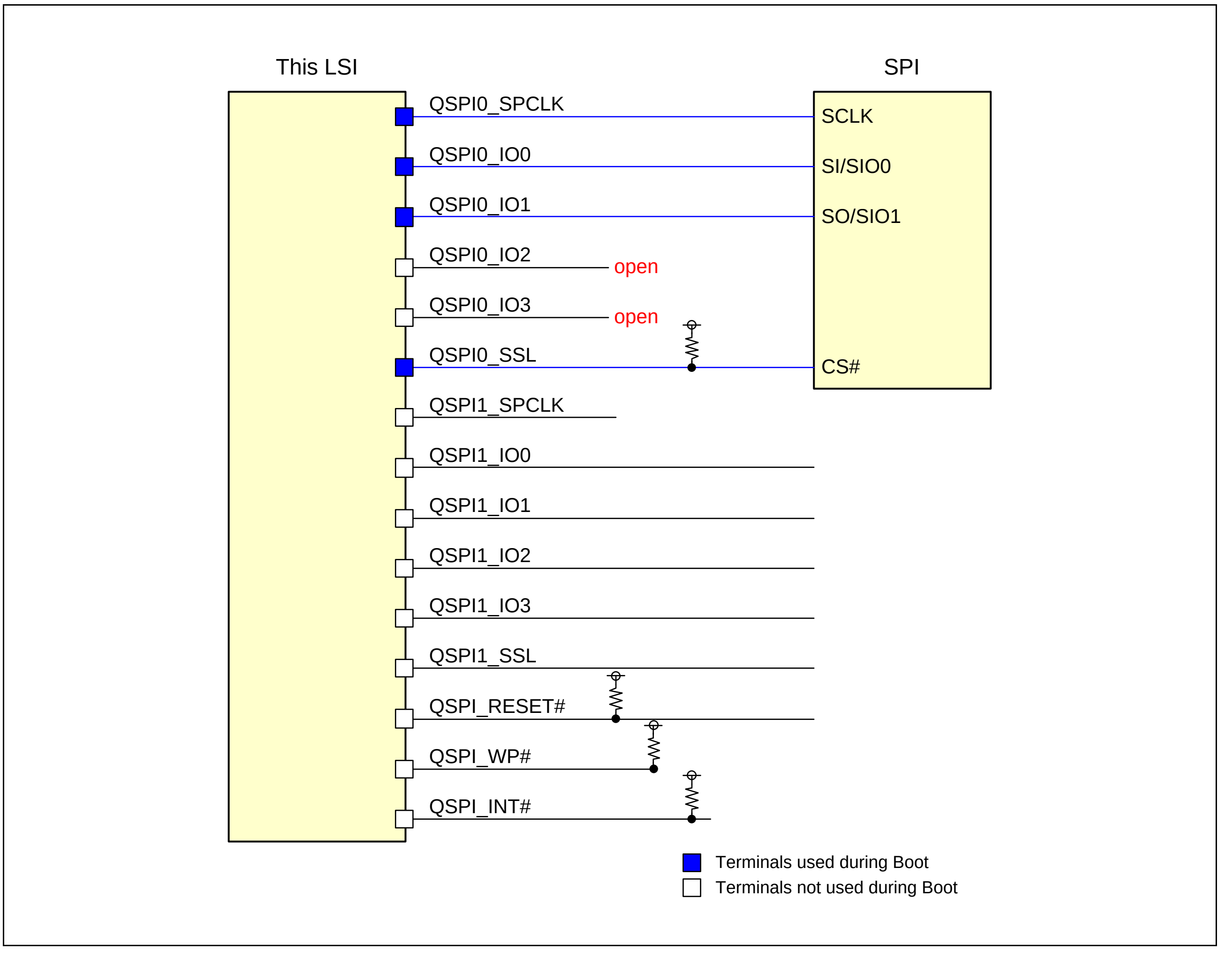


Figure 4.11 Connections with Single SPI Memory in Boot Mode 3

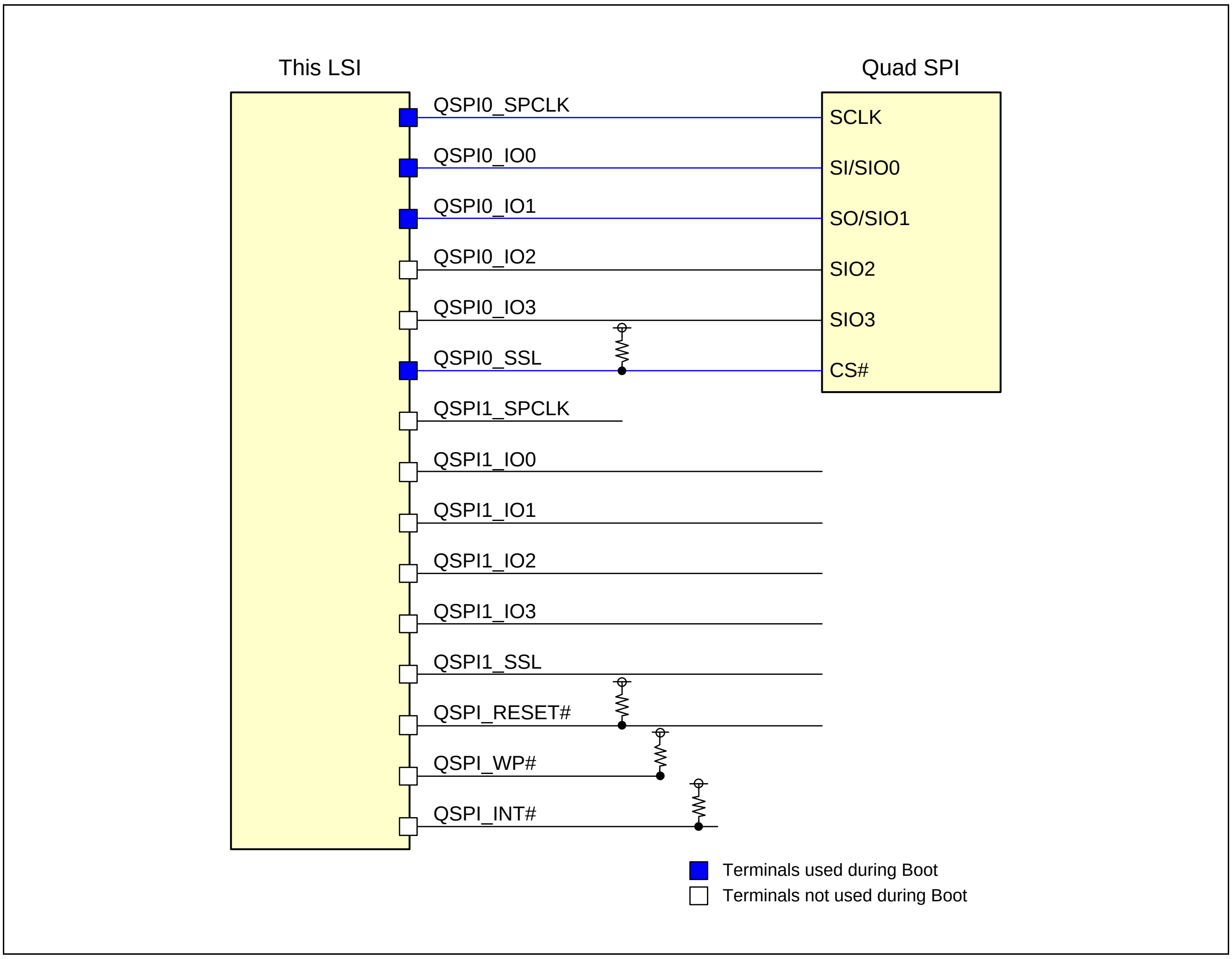


Figure 4.12 Connections with Quad SPI Memory in Boot Mode 3

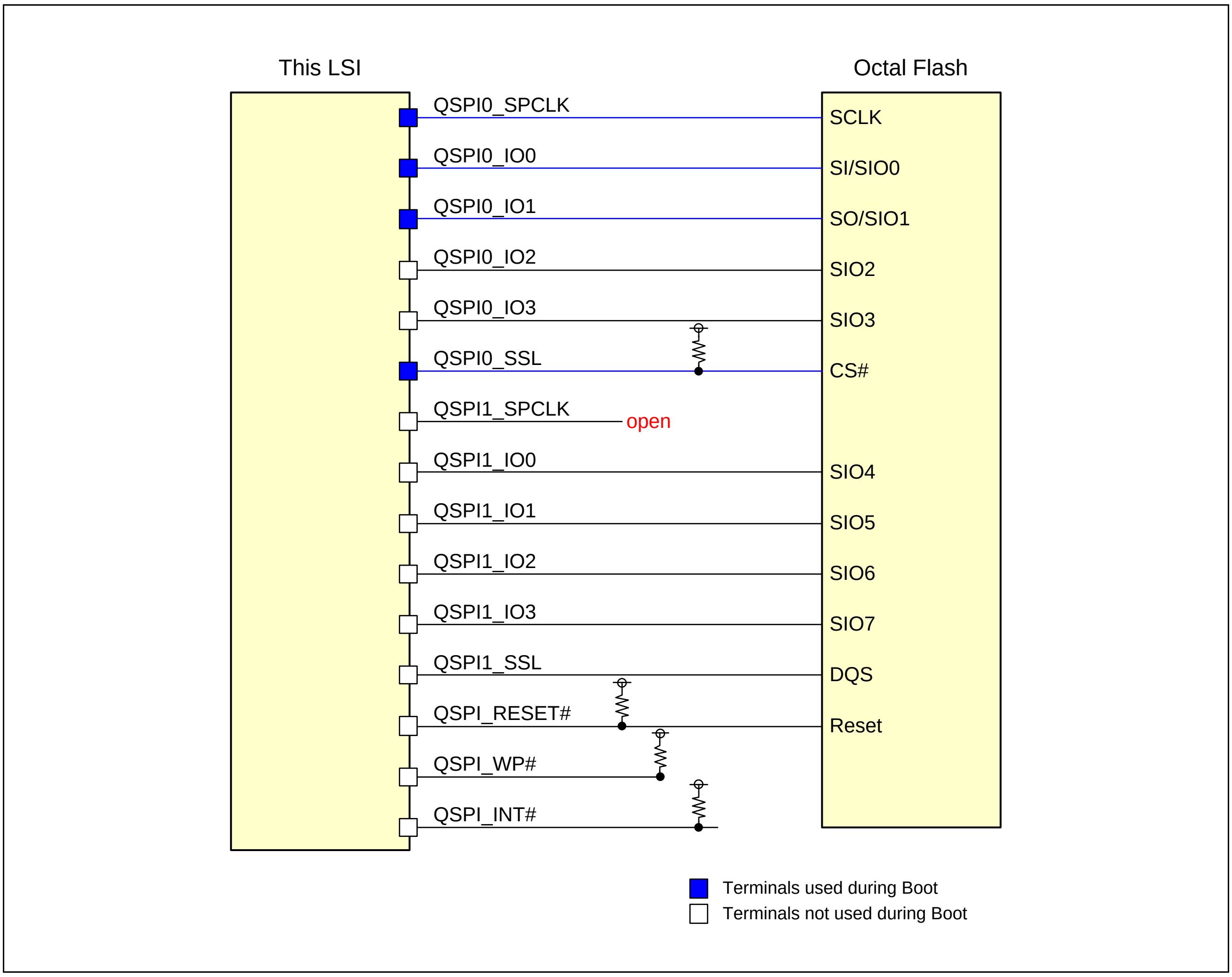


Figure 4.13 Connections with Octal Flash Memory in Boot Mode 3

4.2.4.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the Single, Quad, or Octal flash memory device.

The serial flash memory mounted in advance on the board is used in this mode. The target device operates at 1.8 V and the I/O power of this LSI is also fixed to 1.8 V in this mode. The clock frequency at the startup is set to 16.6 MHz. The LSI is booted up through an SPI handshake with the Single, Quad, or Octal flash memory device.

Upon detecting boot mode 3 according to the value read from the SYSC, the boot program makes the necessary settings of the SPI multi I/O bus controller to access memory. The boot program handles the following steps to control the booting process.

1. The CPG is set up to supply clocks, and the SPI multi I/O bus controller receives the operating clock and is released from the reset state.
2. The necessary information to control the I/O buffers is read from the registers in the SYSC.
   The given information is copied from the OTPC to the SYSC registers in advance.
3. The properties such as the driving ability of the I/O buffers of the signals to be used are set up.
4. After the above setup, the SPI multi I/O bus controller performs a handshake process with the target memory device.
5. When a Quad SPI handshake is required, the user program should control the process.

4.2.4.3 Operation of Booting from 1.8-V Serial Flash Memory

In boot mode 3 (booting from the serial flash memory), this LSI is booted from the loader program in the 1.8-V serial flash memory connected to the SPI multi I/O bus controller (SPIBSC).

The boot program executes the following processing to access the SPI multi I/O bus space.

1. Releasing the SPIBSC from module standby mode
2. Setting up the registers in the necessary peripheral modules (SPIBSC and GPIO)
3. Obtaining the size of the loader program data stored in the area starting from "the start address of the SPI multi I/O bus space (H’0_2000_0000) + offset" through the SPIBSC

After obtaining the size of the loader program data, the SPIBSC transfers the loader program stored in the area starting from "the start address of the SPI multi I/O bus space (H’0_2000_0000) + offset" to the addresses H’0_0001_2000 to H’0_0002_EFFF of the on-chip RAM for the obtained data size.

Then, execution branches to the start address (H’0_0001_2000) of the loader program transferred to the on-chip RAM to execute the loader program that was stored in the serial flash memory.

If the boot program has failed to read data, the boot program enters fail-safe mode (SCIF downloading mode). If the fail-safe processing has failed, execution enters an infinite loop in the on-chip ROM and the boot processing is terminated.

4.2.4.4 Allocation of the Loader Program in the Serial Flash Memory

Figure 4.14 shows the allocation of the loader program size block and loader program in the serial flash memory.

In the loader program size block, the loader program size is stored in the first four bytes and the signature H’AA55 is stored in the last two bytes. **Figure 4.15** shows the structure of the loader program size data. Be sure to store the loader program size in 4-byte little endian.

The loader program is transferred from the area starting from the address “H’0_2000_0000 + H’0200 (offset)” to the addresses H’0_0001_2000 to H’0_0002_EFFF of the on-chip RAM for the obtained loader program size.

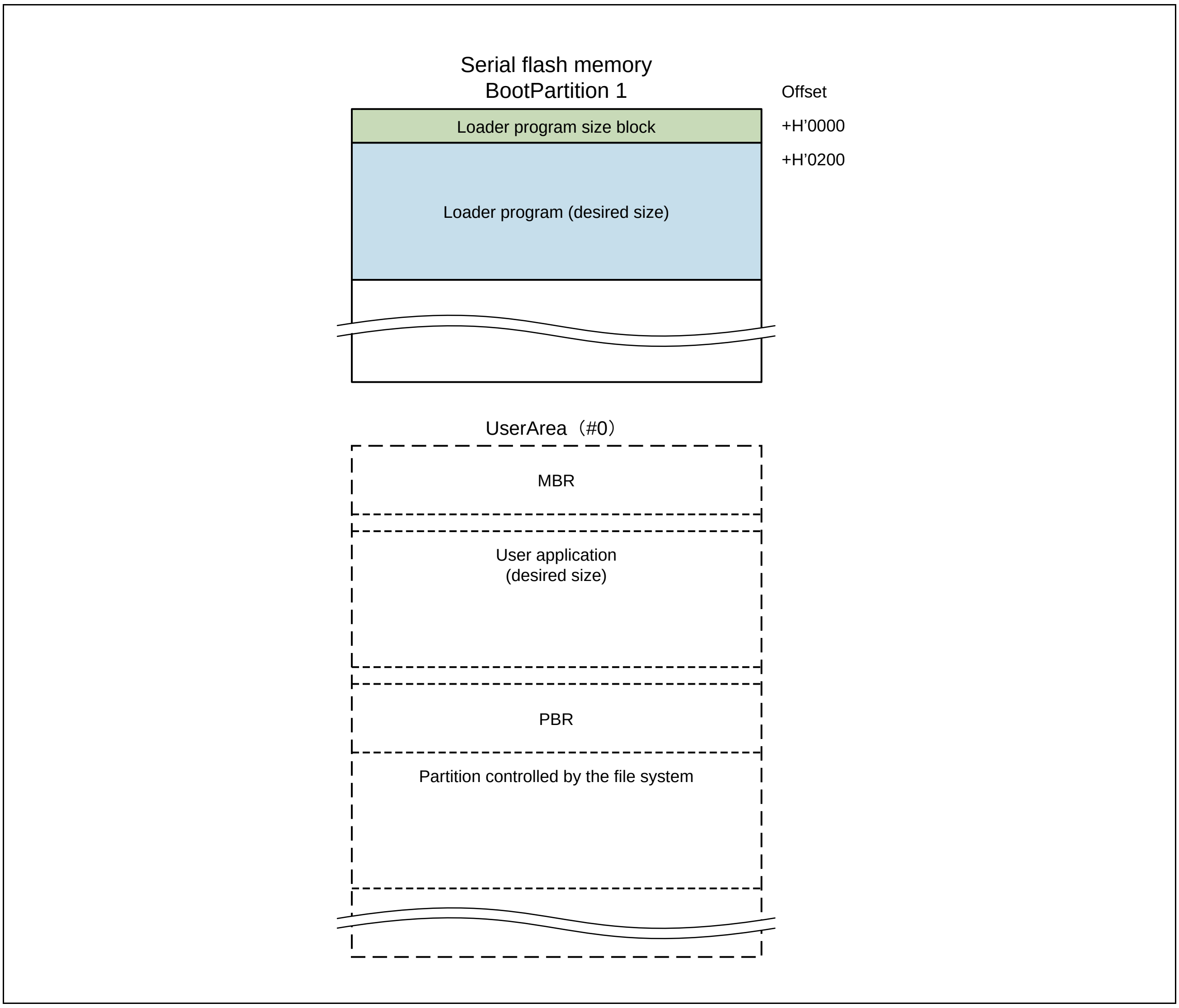


Figure 4.14 Allocation of Loader Program in the Serial Flash Memory Device

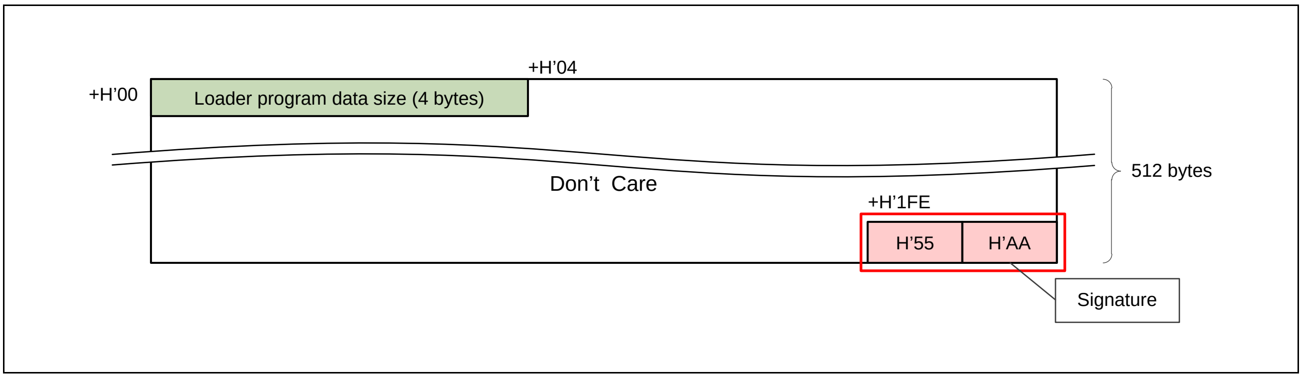


Figure 4.15 Structure of the Loader Program Size Block

4.2.5 Boot Mode 4 (3.3-V Single or Quad Serial Flash Memory)

Table 4.11 shows the interface signals used for connection with the external device in boot mode 4 (3.3-V serial flash memory).

Table 4.11 External Interface Signals Used in Boot Mode 4

Interface Module	Pin Name	I/O	Function	Pin Type
QSPI0	QSPI0_SPCLK	Input/output	QSPI clock	Dedicated pins (3.3 V)
	QSPI0_IO0	Input/output	QSPI data 0	
	QSPI0_IO1	Input/output	QSPI data 1	
	QSPI0_IO2	Input/output	QSPI data 2	
	QSPI0_IO3	Input/output	QSPI data 3	
	QSPI0_SSL	Output	QSPI data 4	
	QSPI1_SPCLK	Output	QSPI clock	
	QSPI1_IO0	Input/output	QSPI data 0	
	QSPI1_IO1	Input/output	QSPI data 1	
	QSPI1_IO2	Input/output	QSPI data 2	
	QSPI1_IO3	Input/output	QSPI data 3	
	QSPI1_SSL	Input/output	QSPI slave select	
	QSPI_RESET#	Output	QSPI reset	
	QSPI_WP#	Output	QSPI write protect	
	QSPI_INT#	Input	QSPI interrupt	

Note: Only interface 0 is used and the pins shaded in gray are not controlled in this mode. To use the shaded pins after booting, control them by the user program.

4.2.5.1 External Connections

Figure 4.16 and **Figure 4.17** show the connections with the 3.3-V Single SPI memory and 3.3-V Quad SPI memory, respectively.

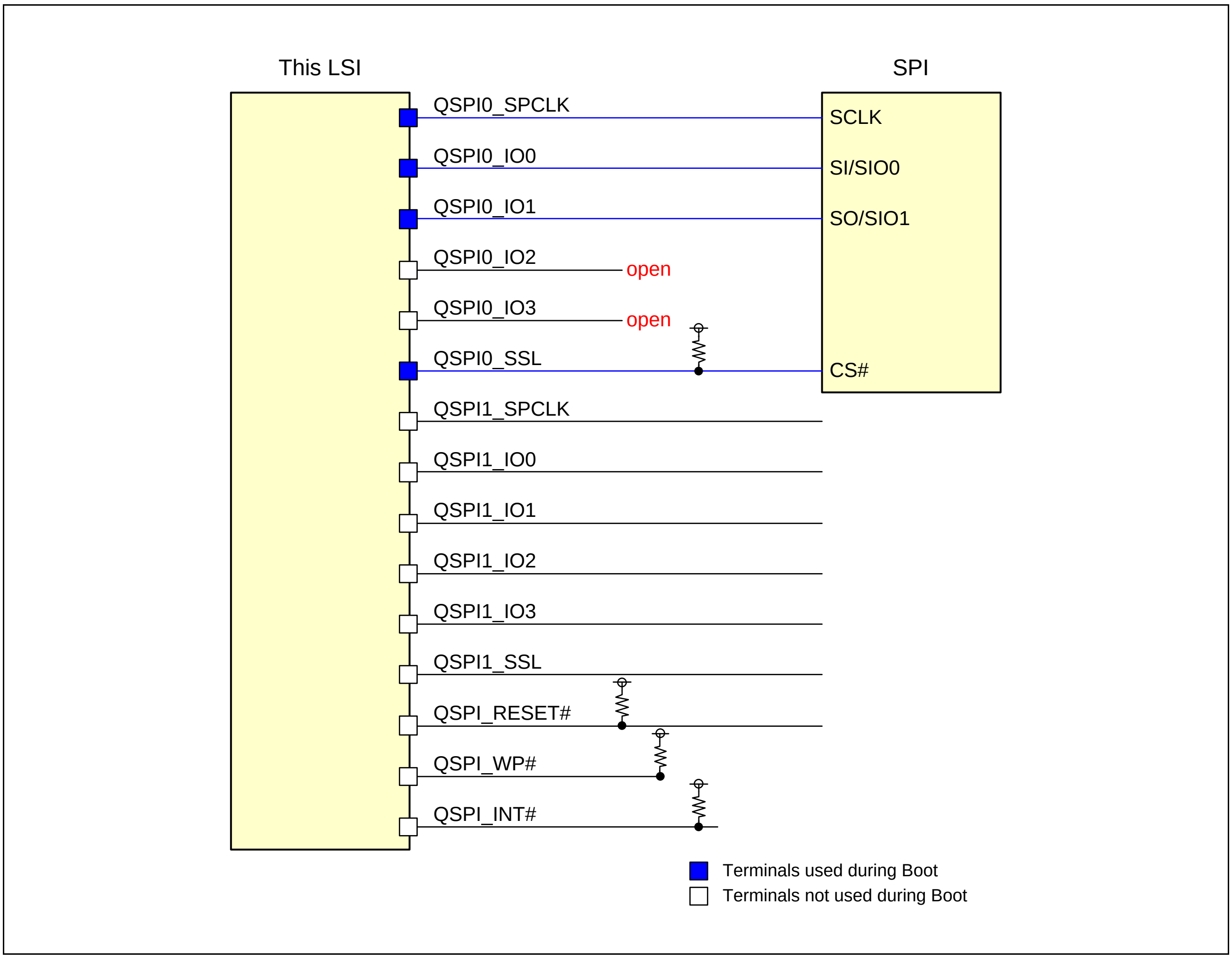


Figure 4.16 Connections with Single SPI Memory in Boot Mode 4

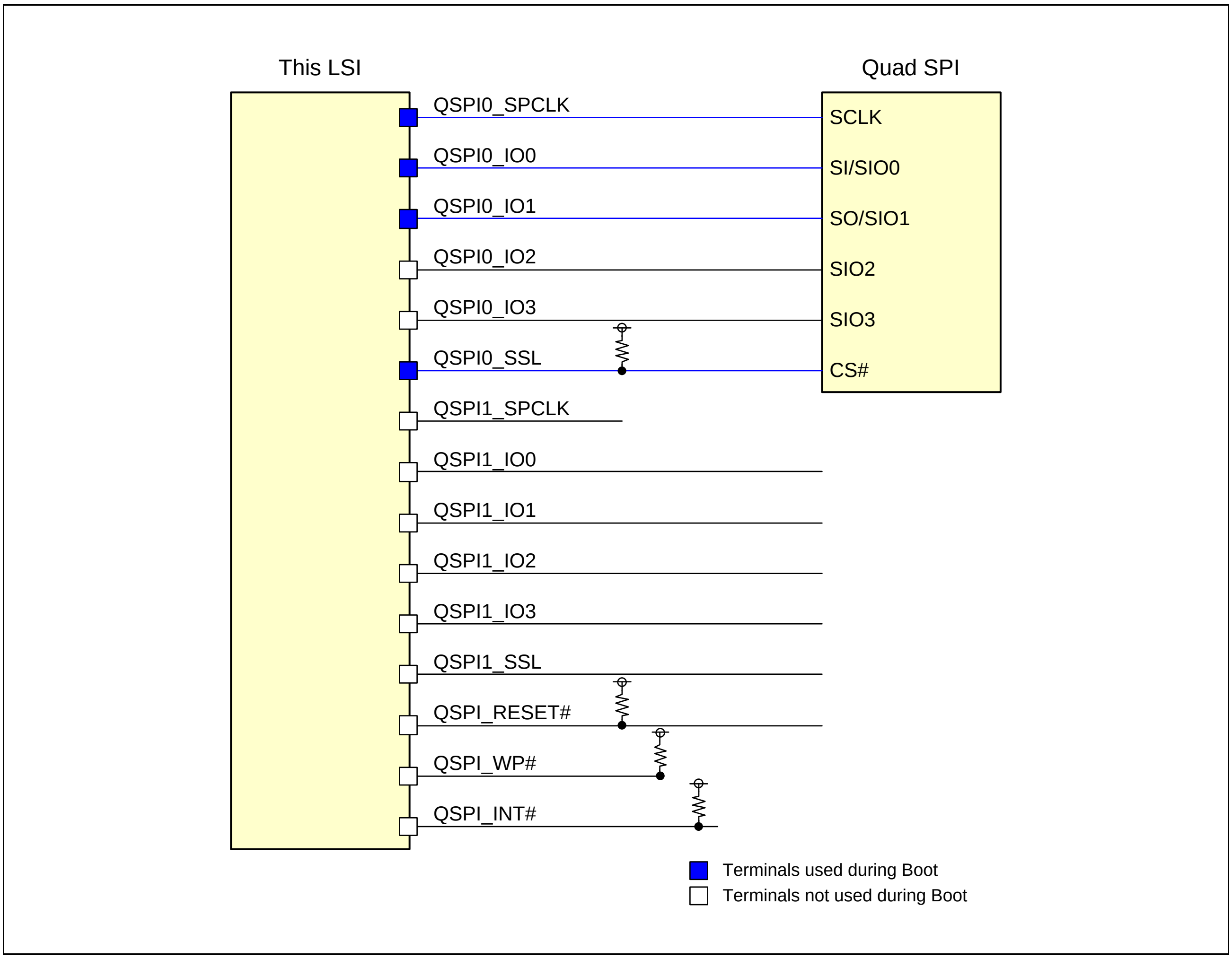


Figure 4.17 Connections with Quad SPI Memory in Boot Mode 4

4.2.5.2 Overview of Operation

This mode is used to boot this LSI from the user program stored in the Single or Quad SPI memory device. The serial flash memory mounted in advance on the board is used in this mode. The target device operates at 3.3 V and the I/O power of this LSI is also fixed to 3.3 V in this mode. The clock frequency at the startup is set to 16.6 MHz.

Upon detecting boot mode 4 according to the value read from the SYSC, the boot program makes the necessary settings of the SPI multi I/O bus controller to access memory. The boot program handles the following steps to control the booting process.

1. The CPG is set up to supply clocks, and the SPI multi I/O bus controller receives the operating clock and is released from the reset state.
2. The necessary information to control the I/O buffers is read from the registers in the SYSC.
   The information is copied from the OTPC to the SYSC registers in advance.
3. The properties such as the driving ability of the I/O buffers of the signals to be used are set up.
4. After the above setup, the SPI multi I/O bus controller performs a handshake process with the target memory device.
5. When a Quad SPI handshake is required, the user program should control the process.

4.2.5.3 Operation of Booting from 3.3-V Serial Flash Memory

Boot mode 4 (booting from 3.3-V serial flash memory) only differs from boot mode 3 (booting from 1.8-V serial flash memory) in the setting of supplied voltage as shown in **Table 4.12**.

Table 4.12 Resources Used by the Boot Program for 3.3-V Serial Flash Memory (Difference from 1.8-V Serial Flash Memory)

Register Name	Address	Initial Value	Value after Booting	Remarks
GPIO_QSPI	H’0_1101_3008	H’0000_0000	H’0000_0000	QSPI_PVDD: 0 = 3.3 V In boot mode 3 or 4, this is automatically set by hardware. The user does not need to set this register.

4.2.5.4 Note

When interface 0 of the SPI multi I/O bus controller is connected with the SPI or QSPI memory, interface 1 can only be used to connect the same type of memory.

4.2.6 Boot Mode 5 (SCIF Downloading)

Table 4.13 shows the interface signals used for connection with the external device in boot mode 5 (SCIF downloading).

Table 4.13 External Interface Signals Used in Boot Mode 5

Interface Module	Pin Name	I/O	Function	Pin Type
SCIF0	SCIF0_SCK	Input/output	SCIF0 serial clock	Multiplexed pins (3.3 V)
	SCIF0_RXD	Input	SCIF0 receive data	
	SCIF0_TXD	Output	SCIF0 transmit data	
	SCIF0_CTS#	Input/output	SCIF0 transmission enable	
	SCIF0_RTS#	Output	SCIF0 transmission request	

4.2.6.1 External Connections

Figure 4.18 shows the external connections for downloading through the SCIF.

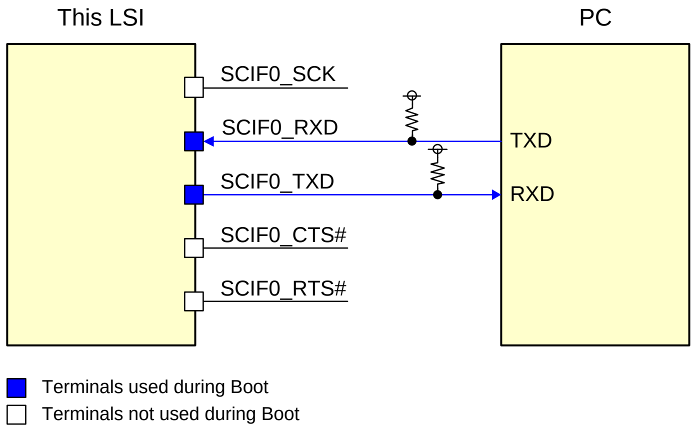


Figure 4.18 Connections for Downloading through the SCIF in Boot Mode 5

4.2.6.2 Overview of Operation

This mode is used to download the user program from the host PC. Upon detecting boot mode 5 according to the value read from the SYSC, the boot program makes the necessary settings of the SCIFA module to access memory. The boot program handles the following steps to control the booting process.

1. The SCIFA module is set up to handle communications as shown below. Asynchronous communications proceed in this mode. The SCIF0_RXD and SCIF0_TXD pins are used.
   Baud rate: 115200 bps
   Data length: 8 bits
   Stop bit: 1 bit
   Flow control: None
   Data format: Motorola S-record
2. The sequence shown in **Figure 4.19** is used to download the user program through the SCIF.

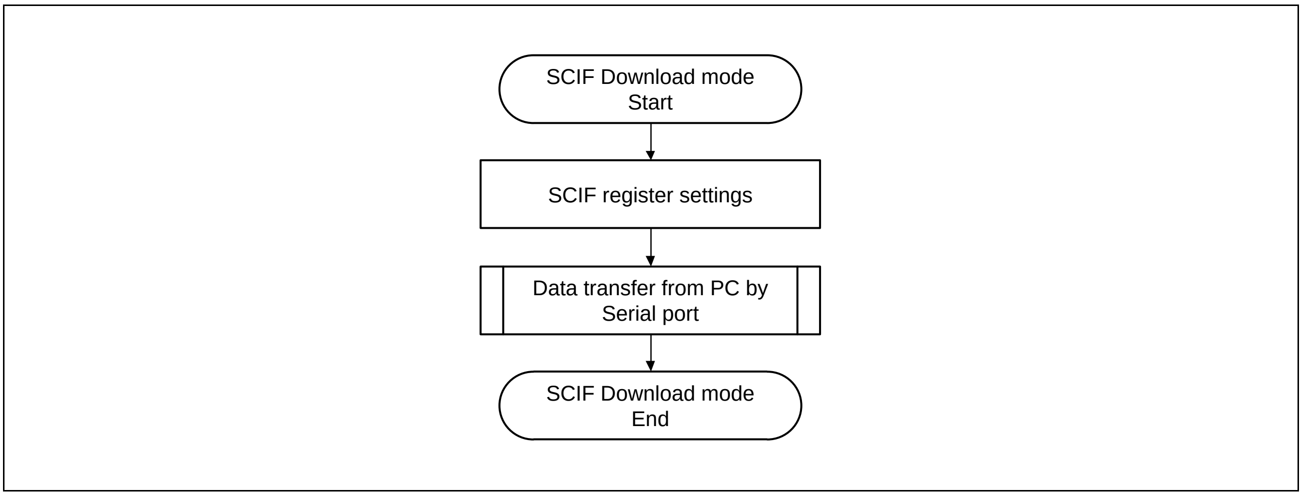


Figure 4.19 Sequence of Downloading through the SCIFA

4.2.6.3 Operation of Booting through SCIF Downloading

In boot mode 5 (booting through SCIF downloading), this LSI is booted from the loader program downloaded from the external host PC through the serial communications interface with FIFO (SCIFA).

The boot program executes the following processing.

1. Setting up the necessary peripheral modules (SCIFA channel 0 and GPIO) and specifying the parameters for communications.

   When the LSI becomes ready for data reception, the boot program outputs the following message to the SCIFA.
   "SCIF Download mode"

   If execution enters boot mode 5 due to the fail-safe processing, the boot program outputs the following message to the SCIFA.
   "SCIF Download mode due to parameter error"

2. Upon receiving the Motorola S0 record sent from the external host PC through the SCIFA, the boot program converts the S3 record to binary data and copies the data to the addresses H'0_0001_2000 to H'0_0002_EFFE (up to 116 Kbytes) of the on-chip RAM.

   The boot program outputs the following message to the SCIFA.
   "-- Load Program to System RAM ---------------"

   Note: The S0 record must always be generated because the boot program requires an S0 record as a trigger for the reception processing.

3. Upon receiving the S7 record, the boot program outputs the following message to the SCIFA and terminates copying to the on-chip RAM. After that, execution branches to the address H'0_0001_2000 of the on-chip RAM.
   "-- Start Boot Program on System RAM ---------"

   If the data in the Motorola S records sent from the external host PC satisfies any of the following error conditions, the corresponding error message is output to the SCIFA and copying to the on-chip RAM is aborted.

   — An S1 or S2 record is received: "Invalid Record Type Error!!! "

   — An address outside the on-chip RAM area is specified: "Address Error!!! "

   — An illegal character code is found: "Invalid Character Error!!! "

   — An illegal byte count is found: "Invalid Byte Count Error!!! "

The data transmission and reception operation in SCIF communications (asynchronous) is shown in **Figure 4.20**.

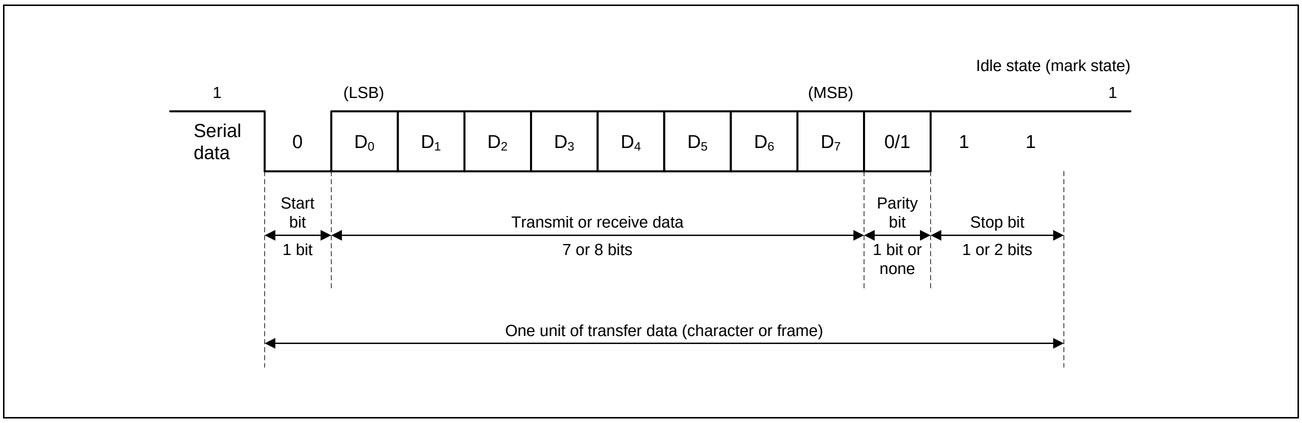
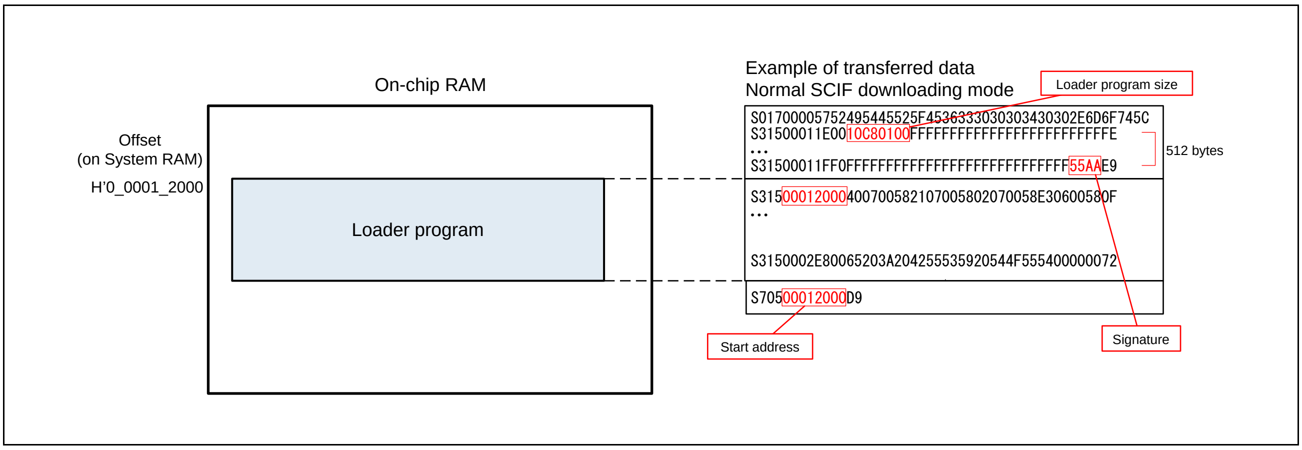


Figure 4.20 Data Transmission and Reception Operation in SCIF Communications

4.2.6.4 Allocation of the Loader Program for SCIF Downloading

Figure 4.21 shows the allocation of the loader program copied to the on-chip RAM.

In the loader program size block (512 bytes), the loader program size is stored in the first four bytes and the signature H’AA55 is stored in the last two bytes. Be sure to store the loader program size in 4-byte little endian.



```
S0170000575249544525F4536333030303430302E6D6F745C
S31500011E0010C80100FFFFFFFFFFFFFFFFFFFFFFFFFE
…
S31500011FF0FFFFFFFFFFFFFFFFFFFFFFFFFFFF55AAE9
S315000120004007005821070058020700 58E30600580F
…
S3150002E80065203A204255535920544F555400000072
S70500012000D9
```

Figure 4.21 Allocation of the Loader Program for SCIF Downloading Copied to the On-chip RAM

4.2.6.5 Note

The pins assigned to the SCIF0_SCK, SCIF0_CTS, and SCIF0_RTS signals, which are not used in this mode, must not be changed from the initial settings (GPIO pins). If such a pin is set to operate as an SCIF0 pin, it becomes an input pin or an input/output pin. In this case, when the pin is externally placed in the Hi-Z state, this LSI cannot communicate with the host PC correctly.

To use the SCIFA for normal serial communications after booting, the SCIF0_SCK, SCIF0_CTS, and SCIF0_RTS pins should be pulled up or down in accordance with the communication mode and the driver software specifications.

5. LSI Internal Bus

5.1 Overview

5.1.1 Features

The bus system of this LSI provides a physical address space of 16 Gbytes (address bus width of 34 bits). The LSI internal bus of this LSI incorporates Arm CoreLink NIC-400, etc., and controls the following bus functions.

Security control:

Security attribute re-setting, Security level determination

Address translation:

34-bit address space access

Interrupt generation:

AXI bus error interrupt generation

Unit state detection:

Slave unit stop state detection

NOTE

The security control function is valid only for secure products.
For information on secure products, please contact our sales.

5.1.2 Block Diagram of LSI Internal Bus

The LSI internal bus of this LSI consists of the ACPU bus, MCPU bus, and system bus. **Figure 5.1** shows the configuration of the buses.

ACPU bus:

A bus connected to Cortex-A55, DDR memory controllers, image processing units, and Storage and Network

MCPU bus:

A bus connected to Cortex-M33 and serial interface units

System bus:

A bus connected to the control registers of each unit

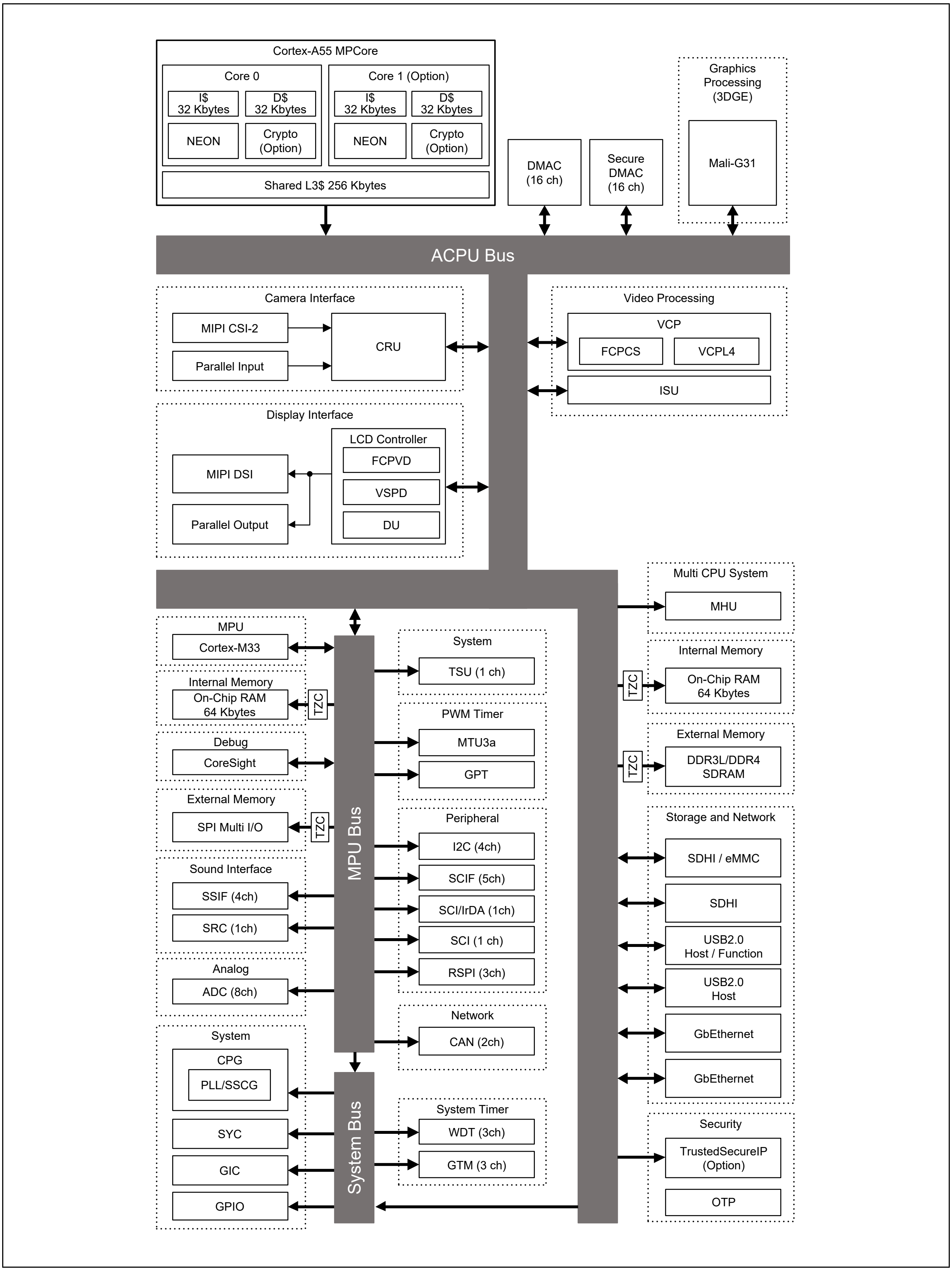


Figure 5.1 Configuration of LSI Internal Bus

5.2 Area Maps

5.2.1 Overall Address Space

Figure 5.2 shows the overall address space of this LSI and **Table 5.1** shows the detailed address space.

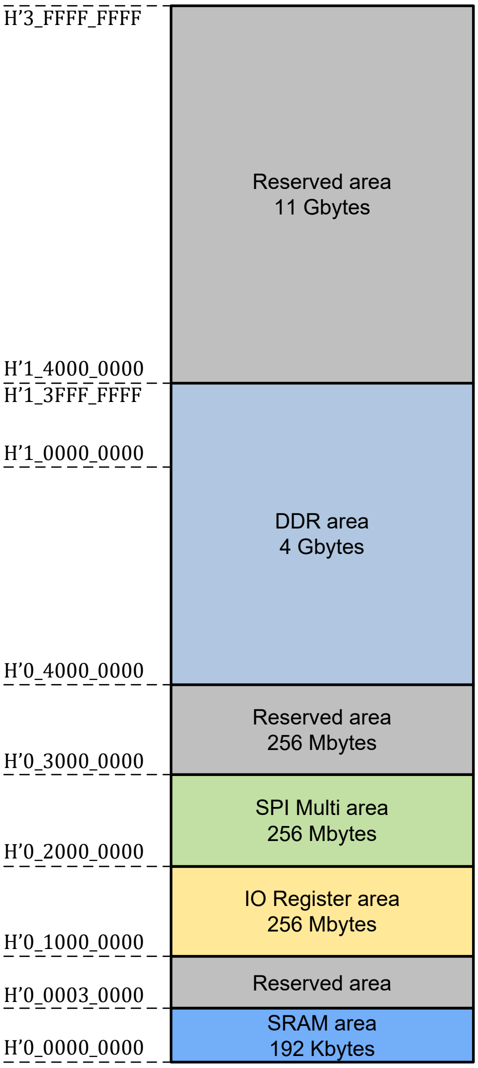


Figure 5.2 Overall Address Space

Some bus master units of this LSI require 34-bit address space access to access an address space of greater than 4 Gbytes in the whole 16-Gbyte address space of this LSI. For details, see **Section 5.4.2.1, 34-Bit Address Space Access**.

The Cortex-M33 address space for this LSI differs from the overall address space. It is the same as the address space of the default memory map of Cortex-M33. For details, see **Section 5.2.2, Cortex-M33 Address Space**.

Table 5.1 Detailed Address Space (1/3)

Start Address	End Address	Size	Space	Remarks
H’1_4000_0000	H’3_FFFF_FFFF	11 Gbytes	Reserved	*1
H’0_4000_0000	H’1_3FFF_FFFF	4 Gbytes	DDR (Memory)	*2
H’0_3000_0000	H’0_3FFF_FFFF	256 Mbytes	Reserved	*1
H’0_2000_0000	H’0_2FFF_FFFF	256 Mbytes	SPI Multi I/O (Memory)	
H’0_1500_0000	H’0_1FFF_FFFF	176 Mbytes	Reserved	*1
H’0_14F0_0000	H’0_14FF_FFFF	1 Mbyte	Reserved	*1
H’0_14C0_0000	H’0_14EF_FFFF	3 Mbytes	Reserved	*1
H’0_14B0_0000	H’0_14BF_FFFF	1 Mbyte	Reserved	*1
H’0_1400_0000	H’0_14AF_FFFF	11 Mbytes	Reserved	*1
H’0_1300_0000	H’0_13FF_FFFF	16 Mbytes	Reserved	*1
H’0_12C0_0000	H’0_12FF_FFFF	4 Mbytes	Reserved	*1
H’0_12B0_0000	H’0_12BF_FFFF	1 Mbyte	Reserved	*1
H’0_12A0_0000	H’0_12AF_FFFF	1 Mbyte	Reserved	*1
H’0_1280_1C00	H’0_129F_FFFF	2041 Kbytes	Reserved	*1
H’0_1280_1800	H’0_1280_1BFF	1 Kbyte	GTM ch2	
H’0_1280_1400	H’0_1280_17FF	1 Kbyte	GTM ch1	
H’0_1280_1000	H’0_1280_13FF	1 Kbyte	GTM ch0	
H’0_1280_0C00	H’0_1280_0FFF	1 Kbyte	WDT CA55 core 1	
H’0_1280_0800	H’0_1280_0BFF	1 Kbyte	WDT CA55 core 0	
H’0_1280_0400	H’0_1280_07FF	1 Kbyte	WDT CM33	
H’0_1280_0000	H’0_1280_03FF	1 Kbyte	Reserved	*1
H’0_1270_0000	H’0_127F_FFFF	1 Mbyte	Reserved	*1
H’0_1240_0000	H’0_126F_FFFF	3 Mbytes	Reserved	*1
H’0_1230_0000	H’0_123F_FFFF	1 Mbyte	Reserved	*1
H’0_1200_0000	H’0_122F_FFFF	3 Mbytes	Reserved	*1
H’0_11F0_0000	H’0_11FF_FFFF	1 Mbyte	Reserved	*1
H’0_11C8_0000	H’0_11EF_FFFF	2560 Kbytes	Reserved	*1
H’0_11C7_0000	H’0_11C7_FFFF	64 Kbytes	USB ch1 Host	
H’0_11C6_0000	H’0_11C6_FFFF	64 Kbytes	USB ch0 (OTG-Func)	
H’0_11C5_0000	H’0_11C5_FFFF	64 Kbytes	USB ch0 (OTG-Host)	
H’0_11C4_0000	H’0_11C4_FFFF	64 Kbytes	USBPHY Control	
H’0_11C3_0000	H’0_11C3_FFFF	64 Kbytes	Ether ch1	
H’0_11C2_0000	H’0_11C2_FFFF	64 Kbytes	Ether ch0	
H’0_11C1_0000	H’0_11C1_FFFF	64 Kbytes	SD ch1	
H’0_11C0_0000	H’0_11C0_FFFF	64 Kbytes	SD ch0	
H’0_11B0_0000	H’0_11BF_FFFF	1 Mbyte	Reserved	*1
H’0_11A0_0000	H’0_11AF_FFFF	1 Mbyte	Reserved	*1
H’0_1190_0000	H’0_119F_FFFF	1 Mbyte	GIC	
H’0_1187_0000	H’0_118F_FFFF	1 Mbyte	Reserved	*1
H’0_1186_0000	H’0_1186_FFFF	64 Kbytes	OTP	
H’0_1185_0000	H’0_1185_FFFF	64 Kbytes	Reserved	*1
H’0_1184_0000	H’0_1184_FFFF	64 Kbytes	Mali-G31	
H’0_1183_0000	H’0_1183_FFFF	64 Kbytes	Non Secure DMAC (DMAC_NS) APB	
H’0_1182_0000	H’0_1182_FFFF	64 Kbytes	Non Secure DMAC (DMAC_NS) AXI	
H’0_1181_0000	H’0_1181_FFFF	64 Kbytes	Secure DMAC (DMAC_S) APB	

Table 5.1 Detailed Address Space (2/3)

Start Address	End Address	Size	Space	Remarks
H’0_1180_0000	H’0_1180_FFFF	64 Kbytes	Secure DMAC (DMAC_S) AXI	
H’0_1170_0000	H’0_117F_FFFF	1 Mbyte	Reserved	*1
H’0_1142_0000	H’0_116F_FFFF	2944 Kbytes	Reserved	*1
H’0_1141_0000	H’0_1141_FFFF	64 Kbytes	DDR (Control Reg)	
H’0_1140_0000	H’0_1140_FFFF	64 Kbytes	DDR (PHY)	
H’0_1130_0000	H’0_113F_FFFF	1 Mbyte	Reserved	*1
H’0_110C_0000	H’0_112F_FFFF	2304 Kbytes	Reserved	*1
H’0_110B_0000	H’0_110B_FFFF	64 Kbytes	IM33 (Interrupt controller)	
H’0_110A_0000	H’0_110A_FFFF	64 Kbytes	IA55 (Interrupt controller)	
H’0_1109_0000	H’0_1109_FFFF	64 Kbytes	SRAM MCPU (Reg)	
H’0_1108_0000	H’0_1108_FFFF	64 Kbytes	SRAM ACPU (Reg)	
H’0_1107_0000	H’0_1107_FFFF	64 Kbytes	TZC (DDR)	
H’0_1106_0000	H’0_1106_FFFF	64 Kbytes	TZC (SPI Multi I/O)	
H’0_1105_0000	H’0_1105_FFFF	64 Kbytes	TZC (SRAM MCPU)	
H’0_1104_0000	H’0_1104_FFFF	64 Kbytes	TZC (SRAM ACPU)	
H’0_1103_0000	H’0_1103_FFFF	64 Kbytes	GPIO	
H’0_1102_0000	H’0_1102_FFFF	64 Kbytes	SYSC	
H’0_1101_0000	H’0_1101_FFFF	64 Kbytes	CPG	
H’0_1100_0000	H’0_1100_FFFF	64 Kbytes	SYC	
H’0_10C0_0000	H’0_10FF_FFFF	4 Mbytes	CST (CoreSight)	
H’0_10B0_0000	H’0_10BF_FFFF	1 Mbyte	Reserved	*1
H’0_108A_0000	H’0_10AF_FFFF	2432 Kbytes	Reserved	*1
H’0_1089_0000	H’0_1089_FFFF	64 Kbytes	DU	
H’0_1088_0000	H’0_1088_FFFF	64 Kbytes	FCPVD	
H’0_1087_0000	H’0_1087_FFFF	64 Kbytes	VSPD	
H’0_1086_0000	H’0_1086_FFFF	64 Kbytes	DSI (Controller)	
H’0_1085_0000	H’0_1085_FFFF	64 Kbytes	DSI (PHY)	
H’0_1084_0000	H’0_1084_FFFF	64 Kbytes	ISU	
H’0_1083_0000	H’0_1083_FFFF	64 Kbytes	CRU	
H’0_1082_0000	H’0_1082_FFFF	64 Kbytes	Reserved	*1
H’0_1081_0000	H’0_1081_FFFF	64 Kbytes	Reserved	*1
H’0_1080_0000	H’0_1080_FFFF	64 Kbytes	Reserved	*1
H’0_1070_0000	H’0_107F_FFFF	1 Mbyte	Reserved	*1
H’0_1060_0000	H’0_106F_FFFF	1 Mbyte	Reserved	*1
H’0_1041_0000	H’0_105F_FFFF	1984 Kbytes	Reserved	*1
H’0_1040_0000	H’0_1040_FFFF	64 Kbytes	MHU	
H’0_1030_0000	H’0_103F_FFFF	1 Mbyte	Reserved	*1
H’0_1020_0000	H’0_102F_FFFF	1 Mbyte	Reserved	*1
H’0_1008_0000	H’0_101F_FFFF	1536 Kbytes	Reserved	*1
H’0_1007_0000	H’0_1007_FFFF	64 Kbytes	SPI Multi I/O (Write Buf)	
H’0_1006_0000	H’0_1006_FFFF	64 Kbytes	SPI Multi I/O (Reg)	
H’0_1005_9800	H’0_1005_FFFF	26 Kbytes	Reserved	*1
H’0_1005_9400	H’0_1005_97FF	1 Kbyte	TSU	
H’0_1005_9000	H’0_1005_93FF	1 Kbyte	ADC	
H’0_1005_8C00	H’0_1005_8FFF	1 Kbyte	I2C ch3	

Table 5.1 Detailed Address Space (3/3)

Start Address	End Address	Size		Space	Remarks
H'0_1005_8800	H'0_1005_8BFF	1	Kbyte	I2C ch2	
H'0_1005_8400	H'0_1005_87FF	1	Kbyte	I2C ch1	
H'0_1005_8000	H'0_1005_83FF	1	Kbyte	I2C ch0	
H'0_1005_0000	H'0_1005_7FFF	32	Kbytes	CANFD	
H'0_1004_D800	H'0_1004_FFFF	10	Kbytes	Reserved	*1
H'0_1004_D400	H'0_1004_D7FF	1	Kbyte	SCI ch1	
H'0_1004_D000	H'0_1004_D3FF	1	Kbyte	SCI ch0	
H'0_1004_CC00	H'0_1004_CFFF	1	Kbyte	IrDA (SCI)	
H'0_1004_C800	H'0_1004_CBFF	1	Kbyte	SCIF ch4	
H'0_1004_C400	H'0_1004_C7FF	1	Kbyte	SCIF ch3	
H'0_1004_C000	H'0_1004_C3FF	1	Kbyte	SCIF ch2	
H'0_1004_BC00	H'0_1004_BFFF	1	Kbyte	SCIF ch1	
H'0_1004_B800	H'0_1004_BBFF	1	Kbyte	SCIF ch0	
H'0_1004_B400	H'0_1004_B7FF	1	Kbyte	RSPI ch2	
H'0_1004_B000	H'0_1004_B3FF	1	Kbyte	RSPI ch1	
H'0_1004_AC00	H'0_1004_AFFF	1	Kbyte	RSPI ch0	
H'0_1004_A800	H'0_1004_ABFF	1	Kbyte	SSIF ch3	
H'0_1004_A400	H'0_1004_A7FF	1	Kbyte	SSIF ch2	
H'0_1004_A000	H'0_1004_A3FF	1	Kbyte	SSIF ch1	
H'0_1004_9C00	H'0_1004_9FFF	1	Kbyte	SSIF ch0	
H'0_1004_9800	H'0_1004_9BFF	1	Kbyte	POE3	
H'0_1004_9400	H'0_1004_97FF	1	Kbyte	POEGD	
H'0_1004_9000	H'0_1004_93FF	1	Kbyte	POEGC	
H'0_1004_8C00	H'0_1004_8FFF	1	Kbyte	POEGB	
H'0_1004_8800	H'0_1004_8BFF	1	Kbyte	POEGA	
H'0_1004_8000	H'0_1004_87FF	2	Kbytes	GPT	
H'0_1004_7000	H'0_1004_7FFF	4	Kbytes	SRC (Reg)	
H'0_1004_0000	H'0_1004_6FFF	28	Kbytes	SRC (Memory)	
H'0_1000_0000	H'0_1003_FFFF	256	Kbytes	MTU3a	
H'0_0003_0000	H'0_0FFF_FFFF	261952	Kbytes	Reserved	*1
H'0_0002_0000	H'0_0002_FFFF	64	Kbytes	SRAM ACPU (Memory)	
H'0_0001_0000	H'0_0001_FFFF	64	Kbytes	SRAM MCPU (Memory)	
H'0_0000_0000	H'0_0000_FFFF	64	Kbytes	Reserved	*1

Note 1. Access to the reserved areas is prohibited.
If access to the reserved area is attempted, incorrect operation may occur.

Note 2. Access to the addresses H'4000_0000 to H'4000_001F (for DDR3L) or H'4000_0000 to H'4000_003F (for DDR4) in the DDR area is prohibited because these are used for DDR training. For details, see **Section 9**, **DDR3L/DDR4 SDRAM Memory Controller (MEMC)**.

5.2.2 Cortex-M33 Address Space

Figure 5.3 shows the default memory map of Cortex-M33 and an overview of the Cortex-M33 address space for this LSI, and **Table 5.2** shows the detailed address space. For the concept of the secure and non-secure states, see **Section 3, System CPU Cortex-M33**.

In this LSI, addresses H’0000_0000 to H’1FFF_FFFF are used as the code (program) area and addresses H’2000_0000 to H’3FFF_FFFF are used as the data area in the 1-Gbyte SRAM area (H’0000_0000 to H’3FFF_FFFF).

In this LSI, the hardware automatically translates the addresses in the Cortex-M33 address space to those in the overall address space of this LSI. Users can therefore write programs using the addresses in the address space shown in **Figure 5.3** and **Table 5.2**, without considering the overall address space.

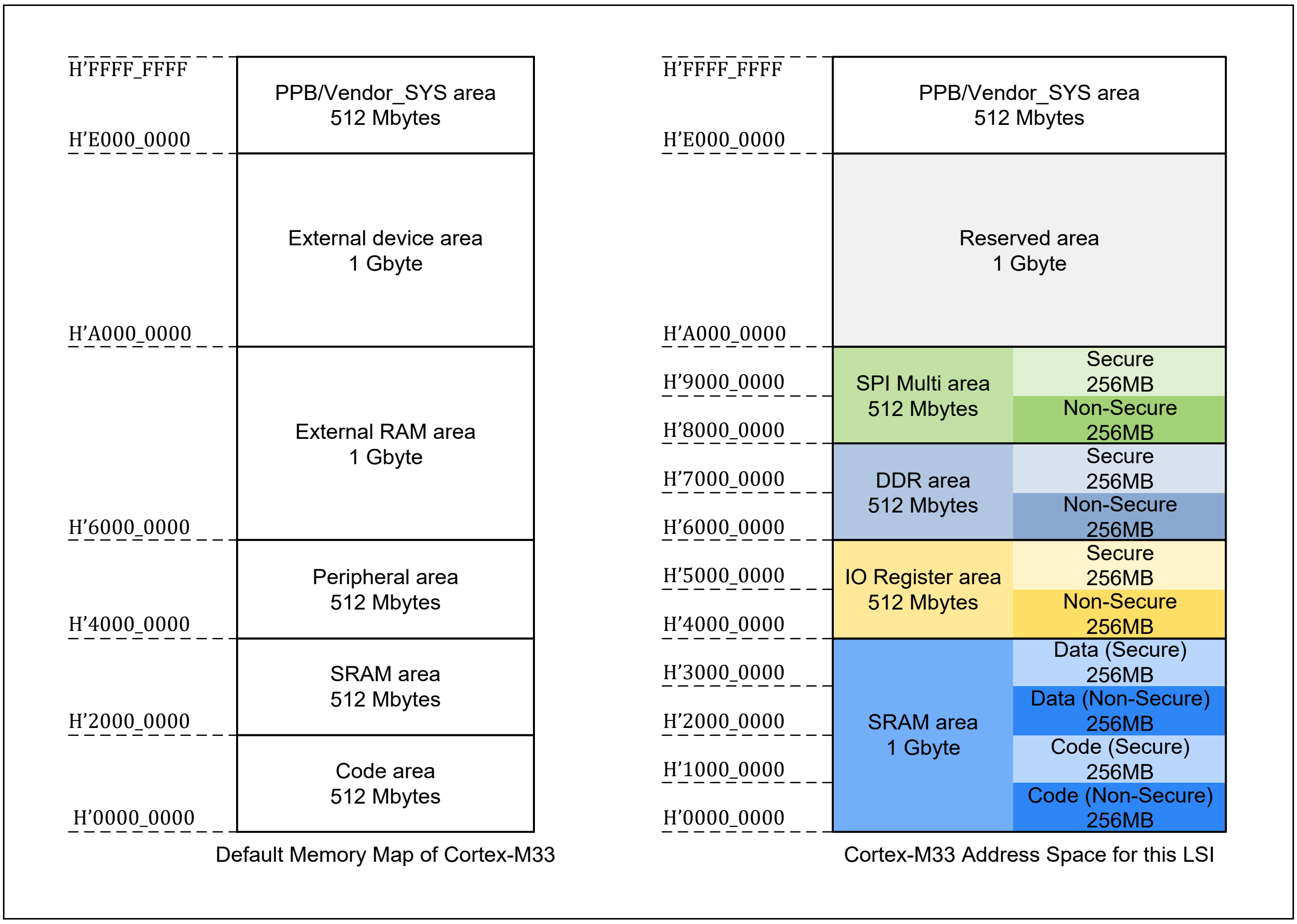


Figure 5.3 Overview of Cortex-M33 Address Space

Table 5.2 Detailed Address Space of Cortex-M33 (1/6)

Start Address	End Address	Size	Space	Remarks
H'E000_0000	H'FFFF_FFFF	512 Mbytes	PPB/Vendor_SYS	*3
H'A000_0000	H'DFFF_FFFF	1 Gbyte	Reserved	*1
H'9000_0000	H'9FFF_FFFF	256 Mbytes	SPI Multi I/O (Memory) (Secure)	
H'8000_0000	H'8FFF_FFFF	256 Mbytes	SPI Multi I/O (Memory) (Non-Secure)	
H'7000_0000	H'7FFF_FFFF	256 Mbytes	DDR (Memory) (Secure)	*2
H'6000_0000	H'6FFF_FFFF	256 Mbytes	DDR (Memory) (Non-Secure)	*2
H'5500_0000	H'5FFF_FFFF	176 Mbytes	Reserved	*1
H'54F0_0000	H'54FF_FFFF	1 Mbyte	Reserved	*1
H'54C0_0000	H'54EF_FFFF	3 Mbytes	Reserved	*1
H'54B0_0000	H'54BF_FFFF	1 Mbyte	Reserved	*1
H'5400_0000	H'54AF_FFFF	11 Mbytes	Reserved	*1
H'5300_0000	H'53FF_FFFF	16 Mbytes	Reserved	*1
H'52C0_0000	H'52FF_FFFF	4 Mbytes	Reserved	*1
H'52B0_0000	H'52BF_FFFF	1 Mbyte	Reserved	*1
H'52A0_0000	H'52AF_FFFF	1 Mbyte	Reserved	*1
H'5280_1C00	H'529F_FFFF	2041 Kbytes	Reserved	*1
H'5280_1800	H'5280_1BFF	1 Kbyte	GTM ch2 (Secure)	
H'5280_1400	H'5280_17FF	1 Kbyte	GTM ch1 (Secure)	
H'5280_1000	H'5280_13FF	1 Kbyte	GTM ch0 (Secure)	
H'5280_0C00	H'5280_0FFF	1 Kbyte	WDT CA55 core 1 (Secure)	
H'5280_0800	H'5280_0BFF	1 Kbyte	WDT CA55 core 0 (Secure)	
H'5280_0400	H'5280_07FF	1 Kbyte	WDT CM33 (Secure)	
H'5280_0000	H'5280_03FF	1 Kbyte	Reserved	*1
H'5270_0000	H'527F_FFFF	1 Mbyte	Reserved	*1
H'5240_0000	H'526F_FFFF	3 Mbytes	Reserved	*1
H'5230_0000	H'523F_FFFF	1 Mbyte	Reserved	*1
H'5200_0000	H'522F_FFFF	3 Mbytes	Reserved	*1
H'51F0_0000	H'51FF_FFFF	1 Mbyte	Reserved	*1
H'51C8_0000	H'51EF_FFFF	2560 Kbytes	Reserved	*1
H'51C7_0000	H'51C7_FFFF	64 Kbytes	USB ch1 Host (Secure)	
H'51C6_0000	H'51C6_FFFF	64 Kbytes	USB ch0 (OTG-Func) (Secure)	
H'51C5_0000	H'51C5_FFFF	64 Kbytes	USB ch0 (OTG-Host) (Secure)	
H'51C4_0000	H'51C4_FFFF	64 Kbytes	USBPHY Control (Secure)	
H'51C3_0000	H'51C3_FFFF	64 Kbytes	Ether ch1 (Secure)	
H'51C2_0000	H'51C2_FFFF	64 Kbytes	Ether ch0 (Secure)	
H'51C1_0000	H'51C1_FFFF	64 Kbytes	SD ch1 (Secure)	
H'51C0_0000	H'51C0_FFFF	64 Kbytes	SD ch0 (Secure)	
H'51B0_0000	H'51BF_FFFF	1 Mbyte	Reserved	*1
H'51A0_0000	H'51AF_FFFF	1 Mbyte	Reserved	*1
H'5190_0000	H'519F_FFFF	1 Mbyte	GIC (Secure)	
H'5187_0000	H'518F_FFFF	1 Mbyte	Reserved	*1
H'5186_0000	H'5186_FFFF	64 Kbytes	OTP (Secure)	
H'5185_0000	H'5185_FFFF	64 Kbytes	Reserved	*1
H'5184_0000	H'5184_FFFF	64 Kbytes	Mali-G31 (Secure)	
H'5183_0000	H'5183_FFFF	64 Kbytes	Non Secure DMAC (DMAC_NS) APB (Secure)	

Table 5.2 Detailed Address Space of Cortex-M33 (2/6)

Start Address	End Address	Size	Space	Remarks
H'5182_0000	H'5182_FFFF	64 Kbytes	Non Secure DMAC (DMAC_NS) AXI (secure)	
H'5181_0000	H'5181_FFFF	64 Kbytes	Secure DMAC (DMAC_S) APB (Secure)	
H'5180_0000	H'5180_FFFF	64 Kbytes	Secure DMAC (DMAC_S) AXI (Secure)	
H'5170_0000	H'517F_FFFF	1 Mbyte	Reserved	*1
H'5142_0000	H'516F_FFFF	2944 Kbytes	Reserved	*1
H'5141_0000	H'5141_FFFF	64 Kbytes	DDR (Control Reg) (Secure)	
H'5140_0000	H'5140_FFFF	64 Kbytes	DDR (PHY) (Secure)	
H'5130_0000	H'513F_FFFF	1 Mbyte	Reserved	*1
H'510C_0000	H'512F_FFFF	2304 Kbytes	Reserved	*1
H'510B_0000	H'510B_FFFF	64 Kbytes	IM33 (Interrupt controller) (Secure)	
H'510A_0000	H'510A_FFFF	64 Kbytes	IA55 (Interrupt controller) (Secure)	
H'5109_0000	H'5109_FFFF	64 Kbytes	SRAM MCPU (Reg) (Secure)	
H'5108_0000	H'5108_FFFF	64 Kbytes	SRAM ACPU (Reg) (Secure)	
H'5107_0000	H'5107_FFFF	64 Kbytes	TZC (DDR) (Secure)	
H'5106_0000	H'5106_FFFF	64 Kbytes	TZC (SPI Multi I/O) (Secure)	
H'5105_0000	H'5105_FFFF	64 Kbytes	TZC (SRAM MCPU) (Secure)	
H'5104_0000	H'5104_FFFF	64 Kbytes	TZC (SRAM ACPU) (Secure)	
H'5103_0000	H'5103_FFFF	64 Kbytes	GPIO (Secure)	
H'5102_0000	H'5102_FFFF	64 Kbytes	SYSC (Secure)	
H'5101_0000	H'5101_FFFF	64 Kbytes	CPG (Secure)	
H'5100_0000	H'5100_FFFF	64 Kbytes	SYC (Secure)	
H'50C0_0000	H'50FF_FFFF	4 Mbytes	CST (CoreSight) (Secure)	
H'50B0_0000	H'50BF_FFFF	1 Mbyte	Reserved	*1
H'508A_0000	H'50AF_FFFF	2432 Kbytes	Reserved	*1
H'5089_0000	H'5089_FFFF	64 Kbytes	DU (Secure)	
H'5088_0000	H'5088_FFFF	64 Kbytes	FCPVD (Secure)	
H'5087_0000	H'5087_FFFF	64 Kbytes	VSPD (Secure)	
H'5086_0000	H'5086_FFFF	64 Kbytes	DSI (Controller) (Secure)	
H'5085_0000	H'5085_FFFF	64 Kbytes	DSI (PHY) (Secure)	
H'5084_0000	H'5084_FFFF	64 Kbytes	ISU (Secure)	
H'5083_0000	H'5083_FFFF	64 Kbytes	CRU (Secure)	
H'5082_0000	H'5082_FFFF	64 Kbytes	Reserved	*1
H'5081_0000	H'5081_FFFF	64 Kbytes	Reserved	*1
H'5080_0000	H'5080_FFFF	64 Kbytes	Reserved	*1
H'5070_0000	H'507F_FFFF	1 Mbyte	Reserved	*1
H'5060_0000	H'506F_FFFF	1 Mbyte	Reserved	*1
H'5041_0000	H'505F_FFFF	1984 Kbytes	Reserved	*1
H'5040_0000	H'5040_FFFF	64 Kbytes	MHU (Secure)	
H'5030_0000	H'503F_FFFF	1 Mbyte	Reserved	*1
H'5020_0000	H'502F_FFFF	1 Mbyte	Reserved	*1
H'5008_0000	H'501F_FFFF	1536 Kbytes	Reserved	*1
H'5007_0000	H'5007_FFFF	64 Kbytes	SPI Multi I/O (Write Buf) (Secure)	
H'5006_0000	H'5006_FFFF	64 Kbytes	SPI Multi I/O (Reg) (Secure)	
H'5005_9800	H'5005_FFFF	26 Kbytes	Reserved	*1
H'5005_9400	H'5005_97FF	1 Kbyte	TSU (Secure)	

Table 5.2 Detailed Address Space of Cortex-M33 (3/6)

Start Address	End Address	Size	Space	Remarks
H'5005_9000	H'5005_93FF	1 Kbyte	ADC (Secure)	
H'5005_8C00	H'5005_8FFF	1 Kbyte	I2C ch3 (Secure)	
H'5005_8800	H'5005_8BFF	1 Kbyte	I2C ch2 (Secure)	
H'5005_8400	H'5005_87FF	1 Kbyte	I2C ch1 (Secure)	
H'5005_8000	H'5005_83FF	1 Kbyte	I2C ch0 (Secure)	
H'5005_0000	H'5005_7FFF	32 Kbytes	CANFD (Secure)	
H'5004_D800	H'5004_FFFF	10 Kbytes	Reserved	*1
H'5004_D400	H'5004_D7FF	1 Kbyte	SCI ch1 (Secure)	
H'5004_D000	H'5004_D3FF	1 Kbyte	SCI ch0 (Secure)	
H'5004_CC00	H'5004_CFFF	1 Kbyte	IrDA (SCI) (Secure)	
H'5004_C800	H'5004_CBFF	1 Kbyte	SCIF ch4 (Secure)	
H'5004_C400	H'5004_C7FF	1 Kbyte	SCIF ch3 (Secure)	
H'5004_C000	H'5004_C3FF	1 Kbyte	SCIF ch2 (Secure)	
H'5004_BC00	H'5004_BFFF	1 Kbyte	SCIF ch1 (Secure)	
H'5004_B800	H'5004_BBFF	1 Kbyte	SCIF ch0 (Secure)	
H'5004_B400	H'5004_B7FF	1 Kbyte	RSPI ch2 (Secure)	
H'5004_B000	H'5004_B3FF	1 Kbyte	RSPI ch1 (Secure)	
H'5004_AC00	H'5004_AFFF	1 Kbyte	RSPI ch0 (Secure)	
H'5004_A800	H'5004_ABFF	1 Kbyte	SSIF ch3 (Secure)	
H'5004_A400	H'5004_A7FF	1 Kbyte	SSIF ch2 (Secure)	
H'5004_A000	H'5004_A3FF	1 Kbyte	SSIF ch1 (Secure)	
H'5004_9C00	H'5004_9FFF	1 Kbyte	SSIF ch0 (Secure)	
H'5004_9800	H'5004_9BFF	1 Kbyte	POE3 (Secure)	
H'5004_9400	H'5004_97FF	1 Kbyte	POEGD (Secure)	
H'5004_9000	H'5004_93FF	1 Kbyte	POEGC (Secure)	
H'5004_8C00	H'5004_8FFF	1 Kbyte	POEGB (Secure)	
H'5004_8800	H'5004_8BFF	1 Kbyte	POEGA (Secure)	
H'5004_8000	H'5004_87FF	2 Kbytes	GPT (Secure)	
H'5004_7000	H'5004_7FFF	4 Kbytes	SRC (Reg) (Secure)	
H'5004_0000	H'5004_6FFF	28 Kbytes	SRC (Memory) (Secure)	
H'5000_0000	H'5003_FFFF	256 Kbytes	MTU3a (Secure)	
H'4500_0000	H'4FFF_FFFF	176 Mbytes	Reserved	*1
H'44F0_0000	H'44FF_FFFF	1 Mbyte	Reserved	*1
H'44C0_0000	H'44EF_FFFF	3 Mbytes	Reserved	*1
H'44B0_0000	H'44BF_FFFF	1 Mbyte	Reserved	*1
H'4400_0000	H'44AF_FFFF	11 Mbytes	Reserved	*1
H'4300_0000	H'43FF_FFFF	16 Mbytes	Reserved	*1
H'42C0_0000	H'42FF_FFFF	4 Mbytes	Reserved	*1
H'42B0_0000	H'42BF_FFFF	1 Mbyte	Reserved	*1
H'42A0_0000	H'42AF_FFFF	1 Mbyte	Reserved	*1
H'4280_1C00	H'429F_FFFF	2041 Kbytes	Reserved	*1
H'4280_1800	H'4280_1BFF	1 Kbyte	GTM ch2 (Non-secure)	
H'4280_1400	H'4280_17FF	1 Kbyte	GTM ch1 (Non-secure)	
H'4280_1000	H'4280_13FF	1 Kbyte	GTM ch0 (Non-secure)	
H'4280_0C00	H'4280_0FFF	1 Kbyte	WDT CA55 core 1 (Non-secure)	

Table 5.2 Detailed Address Space of Cortex-M33 (4/6)

Start Address	End Address	Size	Space	Remarks
H'4280_0800	H'4280_0BFF	1 Kbyte	WDT CA55 core 0 (Non-secure)	
H'4280_0400	H'4280_07FF	1 Kbyte	WDT CM33 (Non-secure)	
H'4280_0000	H'4280_03FF	1 Kbyte	Reserved	*1
H'4270_0000	H'427F_FFFF	1 Mbyte	Reserved	*1
H'4240_0000	H'426F_FFFF	3 Mbytes	Reserved	*1
H'4230_0000	H'423F_FFFF	1 Mbyte	Reserved	*1
H'4200_0000	H'422F_FFFF	3 Mbytes	Reserved	*1
H'41F0_0000	H'41FF_FFFF	1 Mbyte	Reserved	*1
H'41C8_0000	H'41EF_FFFF	2560 Kbytes	Reserved	*1
H'41C7_0000	H'41C7_FFFF	64 Kbytes	USB ch1 Host (Non-secure)	
H'41C6_0000	H'41C6_FFFF	64 Kbytes	USB ch0 (OTG-Func) (Non-secure)	
H'41C5_0000	H'41C5_FFFF	64 Kbytes	USB ch0 (OTG-Host) (Non-secure)	
H'41C4_0000	H'41C4_FFFF	64 Kbytes	USBPHY Control (Non-secure)	
H'41C3_0000	H'41C3_FFFF	64 Kbytes	Ether ch1 (Non-secure)	
H'41C2_0000	H'41C2_FFFF	64 Kbytes	Ether ch0 (Non-secure)	
H'41C1_0000	H'41C1_FFFF	64 Kbytes	SD ch1 (Non-secure)	
H'41C0_0000	H'41C0_FFFF	64 Kbytes	SD ch0 (Non-secure)	
H'41B0_0000	H'41BF_FFFF	1 Mbyte	Reserved	*1
H'41A0_0000	H'41AF_FFFF	1 Mbyte	Reserved	*1
H'4190_0000	H'419F_FFFF	1 Mbyte	GIC (Non-secure)	
H'4187_0000	H'418F_FFFF	1 Mbyte	Reserved	*1
H'4186_0000	H'4186_FFFF	64 Kbytes	OTP (Non-secure)	
H'4185_0000	H'4185_FFFF	64 Kbytes	Reserved	*1
H'4184_0000	H'4184_FFFF	64 Kbytes	Mali-G31 (Non-secure)	
H'4183_0000	H'4183_FFFF	64 Kbytes	Non Secure DMAC (DMAC_NS) APB (Non-Secure)	
H'4182_0000	H'4182_FFFF	64 Kbytes	Non Secure DMAC (DMAC_NS) AXI (Non-secure)	
H'4181_0000	H'4181_FFFF	64 Kbytes	Secure DMAC (DMAC_S) APB (Non-Secure)	
H'4180_0000	H'4180_FFFF	64 Kbytes	Secure DMAC (DMAC_S) AXI (Non-Secure)	
H'4170_0000	H'417F_FFFF	1 Mbyte	Reserved	*1
H'4142_0000	H'416F_FFFF	2944 Kbytes	Reserved	*1
H'4141_0000	H'4141_FFFF	64 Kbytes	DDR (Control Reg) (Non-Secure)	
H'4140_0000	H'4140_FFFF	64 Kbytes	DDR (PHY) (Non-Secure)	
H'4130_0000	H'413F_FFFF	1 Mbyte	Reserved	*1
H'410C_0000	H'412F_FFFF	2304 Kbytes	Reserved	*1
H'410B_0000	H'410B_FFFF	64 Kbytes	IM33 (Interrupt controller) (Non-secure)	
H'410A_0000	H'410A_FFFF	64 Kbytes	IA55 (Interrupt controller) (Non-secure)	
H'4109_0000	H'4109_FFFF	64 Kbytes	SRAM MCPU (Reg) (Non-secure)	
H'4108_0000	H'4108_FFFF	64 Kbytes	SRAM ACPU (Reg) (Non-secure)	
H'4107_0000	H'4107_FFFF	64 Kbytes	TZC (DDR) (Non-secure)	
H'4106_0000	H'4106_FFFF	64 Kbytes	TZC (SPI Multi I/O) (Non-secure)	
H'4105_0000	H'4105_FFFF	64 Kbytes	TZC (SRAM MCPU) (Non-secure)	
H'4104_0000	H'4104_FFFF	64 Kbytes	TZC (SRAM ACPU) (Non-secure)	

Table 5.2 Detailed Address Space of Cortex-M33 (5/6)

Start Address	End Address	Size		Space	Remarks
H'4103_0000	H'4103_FFFF	64	Kbytes	GPIO (Non-secure)	
H'4102_0000	H'4102_FFFF	64	Kbytes	SYSC (Non-secure)	
H'4101_0000	H'4101_FFFF	64	Kbytes	CPG (Non-secure)	
H'4100_0000	H'4100_FFFF	64	Kbytes	SYC (Non-secure)	
H'40C0_0000	H'40FF_FFFF	4	Mbytes	CST (CoreSight)(Non-secure)	
H'40B0_0000	H'40BF_FFFF	1	Mbyte	Reserved	*1
H'408A_0000	H'40AF_FFFF	2432	Kbytes	Reserved	*1
H'4089_0000	H'4089_FFFF	64	Kbytes	DU (Non-secure)	
H'4088_0000	H'4088_FFFF	64	Kbytes	FCPVD (Non-secure)	
H'4087_0000	H'4087_FFFF	64	Kbytes	VSPD (Non-secure)	
H'4086_0000	H'4086_FFFF	64	Kbytes	DSI (Controller) (Non-Secure)	
H'4085_0000	H'4085_FFFF	64	Kbytes	DSI (PHY) (Non-Secure)	
H'4084_0000	H'4084_FFFF	64	Kbytes	ISU (Non-secure)	
H'4083_0000	H'4083_FFFF	64	Kbytes	CRU (Non-secure)	
H'4082_0000	H'4082_FFFF	64	Kbytes	Reserved	*1
H'4081_0000	H'4081_FFFF	64	Kbytes	Reserved	*1
H'4080_0000	H'4080_FFFF	64	Kbytes	Reserved	*1
H'4070_0000	H'407F_FFFF	1	Mbyte	Reserved	*1
H'4060_0000	H'406F_FFFF	1	Mbyte	Reserved	*1
H'4041_0000	H'405F_FFFF	1984	Kbytes	Reserved	*1
H'4040_0000	H'4040_FFFF	64	Kbytes	MHU (Non-secure)	
H'4030_0000	H'403F_FFFF	1	Mbyte	Reserved	*1
H'4020_0000	H'402F_FFFF	1	Mbyte	Reserved	*1
H'4008_0000	H'401F_FFFF	1536	Kbytes	Reserved	*1
H'4007_0000	H'4007_FFFF	64	Kbytes	SPI Multi I/O (Write Buf) (Non-secure)	
H'4006_0000	H'4006_FFFF	64	Kbytes	SPI Multi I/O (Reg) (Non-secure)	
H'4005_9800	H'4005_FFFF	26	Kbytes	Reserved	*1
H'4005_9400	H'4005_97FF	1	Kbyte	TSU (Non-secure)	
H'4005_9000	H'4005_93FF	1	Kbyte	ADC (Non-secure)	
H'4005_8C00	H'4005_8FFF	1	Kbyte	I2C ch3 (Non-secure)	
H'4005_8800	H'4005_8BFF	1	Kbyte	I2C ch2 (Non-secure)	
H'4005_8400	H'4005_87FF	1	Kbyte	I2C ch1 (Non-secure)	
H'4005_8000	H'4005_83FF	1	Kbyte	I2C ch0 (Non-secure)	
H'4005_0000	H'4005_7FFF	32	Kbytes	CANFD (Non-secure)	
H'4004_D800	H'4004_FFFF	10	Kbytes	Reserved	*1
H'4004_D400	H'4004_D7FF	1	Kbyte	SCI ch1 (Non-secure)	
H'4004_D000	H'4004_D3FF	1	Kbyte	SCI ch0 (Non-secure)	
H'4004_CC00	H'4004_CFFF	1	Kbyte	IrDA (SCI) (Non-secure)	
H'4004_C800	H'4004_CBFF	1	Kbyte	SCIF ch4 (Non-secure)	
H'4004_C400	H'4004_C7FF	1	Kbyte	SCIF ch3 (Non-secure)	
H'4004_C000	H'4004_C3FF	1	Kbyte	SCIF ch2 (Non-secure)	
H'4004_BC00	H'4004_BFFF	1	Kbyte	SCIF ch1 (Non-secure)	
H'4004_B800	H'4004_BBFF	1	Kbyte	SCIF ch0 (Non-secure)	
H'4004_B400	H'4004_B7FF	1	Kbyte	RSPI ch2 (Non-secure)	
H'4004_B000	H'4004_B3FF	1	Kbyte	RSPI ch1 (Non-secure)	

Table 5.2 Detailed Address Space of Cortex-M33 (6/6)

Start Address	End Address	Size		Space	Remarks
H'4004_AC00	H'4004_AFFF	1	Kbyte	RSPI ch0 (Non-secure)	
H'4004_A800	H'4004_ABFF	1	Kbyte	SSIF ch3 (Non-secure)	
H'4004_A400	H'4004_A7FF	1	Kbyte	SSIF ch2 (Non-secure)	
H'4004_A000	H'4004_A3FF	1	Kbyte	SSIF ch1 (Non-secure)	
H'4004_9C00	H'4004_9FFF	1	Kbyte	SSIF ch0 (Non-secure)	
H'4004_9800	H'4004_9BFF	1	Kbyte	POE3 (Non-secure)	
H'4004_9400	H'4004_97FF	1	Kbyte	POEGD (Non-secure)	
H'4004_9000	H'4004_93FF	1	Kbyte	POEGC (Non-secure)	
H'4004_8C00	H'4004_8FFF	1	Kbyte	POEGB (Non-secure)	
H'4004_8800	H'4004_8BFF	1	Kbyte	POEGA (Non-secure)	
H'4004_8000	H'4004_87FF	2	Kbytes	GPT (Non-secure)	
H'4004_7000	H'4004_7FFF	4	Kbytes	SRC (Reg) (Non-Secure)	
H'4004_0000	H'4004_6FFF	28	Kbytes	SRC (Memory) (Non-Secure)	
H'4000_0000	H'4003_FFFF	256	Kbytes	MTU3a (Non-secure)	
H'3003_0000	H'3FFF_FFFF	261952	Kbytes	Reserved	*1
H'3002_0000	H'3002_FFFF	64	Kbytes	SRAM ACPU (Memory) (Data, Secure)	
H'3001_0000	H'3001_FFFF	64	Kbytes	SRAM MCPU (Memory) (Data, Secure)	
H'3000_0000	H'3000_FFFF	64	Kbytes	Reserved	*1
H'2003_0000	H'2FFF_FFFF	261952	Kbytes	Reserved	*1
H'2002_0000	H'2002_FFFF	64	Kbytes	SRAM ACPU (Memory) (Data, Non-Secure)	
H'2001_0000	H'2001_FFFF	64	Kbytes	SRAM MCPU (Memory) (Data, Non-Secure)	
H'2000_0000	H'2000_FFFF	64	Kbytes	Reserved	*1
H'1003_0000	H'1FFF_FFFF	261952	Kbytes	Reserved	*1
H'1002_0000	H'1002_FFFF	64	Kbytes	SRAM ACPU (Memory) (Code, Secure)	
H'1001_0000	H'1001_FFFF	64	Kbytes	SRAM MCPU (Memory) (Code, Secure)	
H'1000_0000	H'1000_FFFF	64	Kbytes	Reserved	*1
H'0003_0000	H'0FFF_FFFF	261952	Kbytes	Reserved	*1
H'0002_0000	H'0002_FFFF	64	Kbytes	SRAM ACPU (Memory) (Code, Non-Secure)	
H'0001_0000	H'0001_FFFF	64	Kbytes	SRAM MCPU (Memory) (Code, Non-Secure)	
H'0000_0000	H'0000_FFFF	64	Kbytes	Reserved	*1

Note 1. Access to the reserved areas is prohibited.
If access to the reserved area is attempted, incorrect operation may occur.

Note 2. Access to the addresses H'6000_0000 to H'6000_001F and H'7000_0000 to H'7000_001F (for DDR3L) or H'6000_0000 to H'6000_003F and H'7000_0000 to H'7000_003F (for DDR4) in the DDR area is prohibited because these are used for DDR training. For details, see **Section 9, DDR3L/DDR4 SDRAM Memory Controller (MEMC)**.

Note 3. For details on the PPB/Vendor_SYS area, refer to “Arm Cortex-M33 Processor Technical Reference Manual”.

5.3 Accessible Areas

In the bus system of this LSI, each bus master unit can only access the areas that are used for register access or data transfer. **Table 5.3** shows the areas that can be accessed from each master.

The register area of TZC can be accessed only when Secure (AxPROT[1] = 0).

Table 5.3 Accessible Areas (1/2)

	Master Unit													
Slave Unit	Cortex-A55	Mali-G31	DMAC_S	DMAC_NS	Cortex-M33 [C,S-AHB]	CoreSight [AXI-AP]	CoreSight [ETR]	SDHI (ch0, ch1)	Gether (ch0, ch1)	USB2.0 (ch0, ch1)	CRU [Video, Statistics]	FCPVD	DSI (Controller)	ISU
DDR (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MHU	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SRAM ACPU (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SRAM MCPU (Memory)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SPI Multi I/O (Memory/Reg)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
I2C (ch0 to ch3)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
CANFD	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
ADC	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
TSU	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
POEGA	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
POEGB	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
POEGC	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
POEGD	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
GPT	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SRC (Memory/Reg)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
POE3	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
MTU3a	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SSIF (ch0 to ch3)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
RSPI (ch0 to ch2)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SCIF (ch0 to ch4)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
IrDA (SCI)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SCI (ch0, ch1)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
WDT (ch0 to ch2)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
GTM (ch0 to ch2)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
OTP	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
Mali-G31	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
DMAC_S	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
DMAC_NS	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SDHI (ch0, ch1)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
Ether (ch0, ch1)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
USBPHY Control	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
USB ch0 (OTG-Host)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×

Table 5.3 Accessible Areas (2/2)

Slave Unit	Master Unit													
	Cortex-A55	Mali-G31	DMAC_S	DMAC_NS	Cortex-M33 [C,S-AHB]	CoreSight [AXI-AP]	CoreSight [ETR]	SDHI (ch0, ch1)	Gether (ch0, ch1)	USB2.0 (ch0, ch1)	CRU [Video, Statistics]	FCPVD	DSI (Controler)	ISU
USB ch0 (OTG-Func)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
USB ch1 Host	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
CRU	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
ISU	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
FCPVD	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
VSPD	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
DU	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
DSI (Controller)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
DSI (PHY)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
CoreSight	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SYSC	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SRAM ACPU (Reg)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
SRAM MCPU (Reg)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
TZC (SRAM ACPU)	✓	×	✓	×*1	✓	✓	×	×	×	×	×	×	×	×
TZC (SRAM MCPU)	✓	×	✓	×*1	✓	✓	×	×	×	×	×	×	×	×
TZC (SPI Multi I/O)	✓	×	✓	×*1	✓	✓	×	×	×	×	×	×	×	×
TZC (DDR)	✓	×	✓	×*1	✓	✓	×	×	×	×	×	×	×	×
SYC	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
GPIO	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
CPG	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
IA55 (Interrupt controller)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
IM33 (Interrupt controller)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
GIC	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
DDR (Control Reg)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×
DDR (PHY)	✓	×	✓	✓	✓	✓	×	×	×	×	×	×	×	×

Note 1. Access to TZC-400 registers is prohibited because DMAC_NS is always fixed to the Non-secure state and unprivileged mode by control signals from the SYSC.

Remarks: ✓: Accessible path, ×: Prohibited path

Note: Usage of paths marked with × in the table is prohibited.
If access to such a path is attempted, incorrect operation may occur.

5.4 Bus System Control

5.4.1 Security Control

5.4.1.1 Re-Setting the Security Attribute Output from Bus Masters

This facility is for re-setting the security attributes of bus transactions that are by bus master units. Specifically, use the master access control register (SYS_MSTACCCTLn: n = 0 to 4) to re-set the security attributes. For details on the SYS_MSTACCCTLn register, see **Section 6.3, Register Descriptions**.

NOTE

The bus transaction signals ARPROT[1:0] and AWPROT[1:0] indicate the security attributes, which can be used to protect against illicit transactions.

- ARPROT[0] and AWPROT[0] being 0 or 1 respectively indicate non-privileged access or privileged access.
- ARPROT[1] and AWPROT[1] being 0 or 1 respectively indicate secure access or non-secure access.

The ARPROT and AWPROT signals are prescribed in as part of the AXI protocol. For details, see the AMBA AXI and ACE Protocol Specifications published by Arm Ltd.

Use the SYS_MSTACCCTLn register to make the following settings.

- Selection of security attribute source

  Selection of whether to use the unchanged values of ARPROT[1:0] or AWPROT[1:0] of the ARPROT[2:0] or AWPROT[2:0] signals, which are output from the bus master units, or to re-set ARPROT[1:0] or AWPROT[1:0] according to the settings of the SYS_MSTACCCTL register
- Value to be re-set for ARPROT[1:0] or AWPROT[1:0]

Table 5.4 shows the relation between the registers and bits for use in re-setting the security attributes and the target masters for control.

Table 5.4 List of Register Bits for Use in Re-setting of Security Attributes

Target Master for Control*1	Control Register*2	Write Access Control Bits*3,*4			Read Access Control Bits*3,*5		
		Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source	Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source
		AWPROT[0]	AWPROT[1]		ARPROT[0]	ARPROT[1]	
Secure DMAC	SYS_MSTACCCTL0	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
Non-Secure DMAC*6	SYS_MSTACCCTL0	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
3DGE	SYS_MSTACCCTL0	Bit 16: AWPU	Bit 17: AWNS	Bit 19: AWSEL	Bit 20: ARPU	Bit 21: ARNS	Bit 23: ARSEL
SDHI/eMMC ch0	SYS_MSTACCCTL1	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
SDHI ch1	SYS_MSTACCCTL1	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
GbEthernet ch0	SYS_MSTACCCTL1	Bit 16: AWPU	bit17: AWNS	Bit 19: AWSEL	Bit 20: ARPU	Bit 21: ARNS	Bit 23: ARSEL
GbEthernet ch1	SYS_MSTACCCTL1	Bit 24: AWPU	Bit 25: AWNS	Bit 27: AWSEL	Bit 28: ARPU	Bit 29: ARNS	Bit 31: ARSEL
USB2.0 ch0 Host*7	SYS_MSTACCCTL2	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
USB2.0 ch1 Host*7	SYS_MSTACCCTL2	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
USB2.0 ch0 Function*7	SYS_MSTACCCTL2	Bit 16: AWPU	Bit 17: AWNS	Bit 19: AWSEL	Bit 20: ARPU	Bit 21: ARNS	Bit 23: ARSEL
VCPL4	SYS_MSTACCCTL3	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
LCDC	SYS_MSTACCCTL3	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
MIPI DSI	SYS_MSTACCCTL3	Bit 16: AWPU	Bit 17: AWNS	Bit 19: AWSEL	Bit 20: ARPU	Bit 21: ARNS	Bit 23: ARSEL
ISU*8	SYS_MSTACCCTL4	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
CRU (Video)*8	SYS_MSTACCCTL4	Bit 16: AWPU	Bit 17: AWNS	Bit 19: AWSEL			

Note 1. Re-setting of security attributes for the Cortex-A55, Cortex-M33, and CoreSight bus masters is not possible.

Note 2. See **Section 6.3, Register Descriptions** for details on the registers.

Note 3. See **Section 6.3, Register Descriptions** for details on the bits.

Note 4. The values set in AWPU and AWNS are effective when the setting of the corresponding AWSEL bit is 1. For details, see the specifications of the registers in **Section 6.3, Register Descriptions**.

Note 5. The values set in ARPU and ARNS are effective when the setting of the corresponding ARSEL bit is 1. For details, see the specifications of the registers in **Section 6.3, Register Descriptions**.

Note 6. AWPU and ARPU, AWNS and ARNS, and AWSEL and ARSEL for use in control of the non-secure DMAC are respectively fixed to 1, 1, 0, 0, and 1, 1. Accordingly, access to the non-secure DMAC is always non-secure and non-privileged.

Note 7. When the setting of AWSEL for use in control of the USB2.0 module is 0, the values in AWPROT[0] and ARPROT[0] are the same as the value of HPROT[1] output from the USB2.0 module, and the values of AWPROT[1] and ARPROT[1] are always 1. For control of HPROT[1] in the USB2.0 module, see **Section 32A, USB 2.0 Host Module**. The HPROT signal is prescribed in the HB-Lite protocol. For details, see the AMBA 3 AHB-Lite Protocol Specification published by Arm Ltd.

Note 8. AWSEL and ARSEL for use in control of the ISU and CRU (video) should be set to 1. Set AWPU, AWNS, ARPU, and ARNS for the desired security attributes. Operation is not guaranteed if AWSEL and ARSEL are set to 0. For details, see the specifications of the registers in **Section 6.3, Register Descriptions**.

5.4.1.2 Determining the Security Levels of Bus Slaves

This facility is for enabling or disabling access to a bus slave unit by comparing the security attribute that is input to the bus slave unit in a bus transaction with the security setting of the given slave. The security level can be set in one of the following ways. The applicable way depends on the slave.

- Setting by using the SL[1:0] bits in the slave control register (SYS_SLVACCCTLn: n = 0 to 8, 10, 12 to 14)
- Setting by using the nsaid_wr_en[31:16] and nsaid_rd_en[15:0] bits in the region ID access register (REGION_ID_ACCESS_<n>: n = 0 to 2) of the TrustZone address space controller (TZC). For details on the TZC, see **Section 13, TrustZone Address Space Controller (TZC)**.

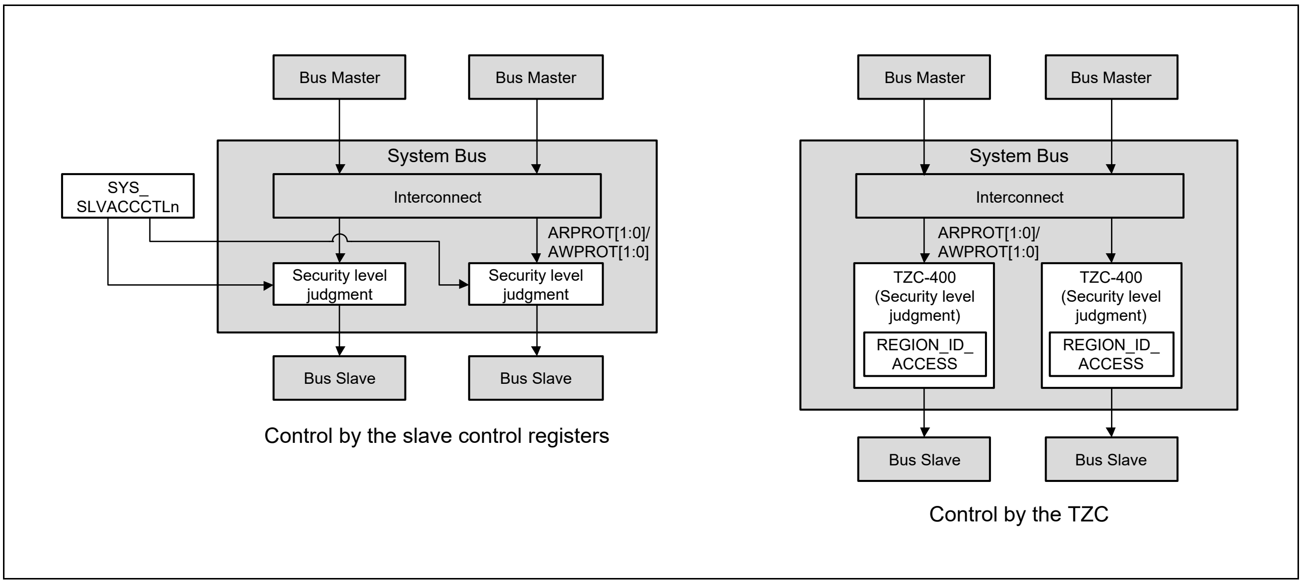


Figure 5.4 Security level determination ways

NOTE

The bus transaction signals ARPROT[1:0] and AWPROT[1:0] indicate the security attributes, which can be used to protect against illicit transactions.

- ARPROT[0] and AWPROT[0] being 0 or 1 respectively indicate non-privileged access or privileged access.
- ARPROT[1] and AWPROT[1] being 0 or 1 respectively indicate secure access or non-secure access.

The ARPROT and AWPROT signals are prescribed in as part of the AXI protocol. For details, see the AMBA AXI and ACE Protocol Specifications published by Arm Ltd.

Table 5.5 shows the correspondence between the settings of the slave control register and the security levels for input transactions. **Table 5.6** shows the correspondence between the settings of the region ID access register of the TZC and the security levels for input bus transactions.

If comparison indicates that access is disabled, a bus error is generated in response to the corresponding attempted bus transaction. Stopping of a slave unit is detected by reference to the module stop state (MSTOP) setting in the CPG. Attempted access to a slave unit in the module stop state leads to a bus error.

Table 5.5 Security Levels Set by Using the Slave Control Register

Setting of SL[1:0]	Security Attribute for Input Bus Transactions			
	Non-privileged Non-secure	Privileged Non-secure	Non-privileged Secure	Privileged Secure
00	Access allowed	Access allowed	Access allowed	Access allowed
01	Access not allowed	Access allowed	Access allowed	Access allowed
10	Access not allowed	Access not allowed	Access allowed	Access allowed
11	Access not allowed	Access not allowed	Access not allowed	Access allowed

Table 5.6 Security Levels Set by Using the TZC

Setting of nsaid_wr_en[31:16] and nsaid_rd_en[15:0]	Security Attribute for Input Bus Transactions			
	Non-privileged Non-secure	Privileged Non-secure	Non-privileged Secure	Privileged Secure
H’000F	Access allowed	Access allowed	Access allowed	Access allowed
H’000B	Access not allowed	Access allowed	Access allowed	Access allowed
H’0003	Access not allowed	Access not allowed	Access allowed	Access allowed
H’0002	Access not allowed	Access not allowed	Access not allowed	Access allowed

Table 5.7 lists the bus slaves for which the security levels are determined by using the slave control registers (SYS_SLVACCCTLn: n = 0 to 8, 10, 12 to 14). **Table 5.8** lists the bus slaves for which security levels are determined by using the region ID access registers (REGION_ID_ACCESS_<n>) of the TZC.

Table 5.7 Bus Slaves for which the Security Levels are Determined by Using the Slave Control Registers (1/2)

Target Bus Slave for Control*[1]	Control Register*[2]	SL[1:0] Allocation Bits*[3]
SRAM ACPU (Reg)	SYS_SLVACCCTL0	Bits [1:0]
SRAM MCPU (Reg)	SYS_SLVACCCTL0	Bits [3:2]
TZC (SRAM ACPU) (Reg)	SYS_SLVACCCTL1	Bits [1:0]
TZC (SRAM MCPU) (Reg)	SYS_SLVACCCTL1	Bits [3:2]
TZC (SPI Multi I/O) (Reg)	SYS_SLVACCCTL1	Bits [5:4]
TZC (DDR) (Reg)	SYS_SLVACCCTL1	Bits [7:6]
CoreSight	SYS_SLVACCCTL1	Bits [11:10]
CPG	SYS_SLVACCCTL1	Bits [13:12]
SYSC	SYS_SLVACCCTL1	Bits [15:14]
SYC	SYS_SLVACCCTL1	Bits [17:16]
GIC	SYS_SLVACCCTL1	Bits [19:18]
IA55/IM33	SYS_SLVACCCTL1	Bits [21:20]
GPIO	SYS_SLVACCCTL1	Bits [23:22]
MHU	SYS_SLVACCCTL1	Bits [25:24]
Secure DMAC	SYS_SLVACCCTL1	Bits [27:26]
Non-Secure DMAC	SYS_SLVACCCTL1	Bits [29:28]
GTM (channel 0)	SYS_SLVACCCTL2	Bits [1:0]
GTM (channel 1)	SYS_SLVACCCTL2	Bits [3:2]
GTM (channel 2)	SYS_SLVACCCTL2	Bits [5:4]
WDT (channel 0)	SYS_SLVACCCTL2	Bits [7:6]
WDT (channel 1)	SYS_SLVACCCTL2	Bits [9:8]
WDT (channel 2)	SYS_SLVACCCTL2	Bits [11:10]

Table 5.7 Bus Slaves for which the Security Levels are Determined by Using the Slave Control Registers (2/2)

Target Bus Slave for Control*1	Control Register*2	SL[1:0] Allocation Bits*3
MTU3a	SYS_SLVACCCTL2	Bits [15:14]
POE3	SYS_SLVACCCTL2	Bits [17:16]
GPT	SYS_SLVACCCTL2	Bits [19:18]
POEG	SYS_SLVACCCTL2	Bits [21:20]
DDR3L/4 Controller (Reg)	SYS_SLVACCCTL2	Bits [23:22]
3DGE	SYS_SLVACCCTL3	Bits [1:0]
VCPL4	SYS_SLVACCCTL3	Bits [3:2]
CRU	SYS_SLVACCCTL3	Bits [5:4]
ISU	SYS_SLVACCCTL3	Bits [7:6]
MIPI DSI (PHY)	SYS_SLVACCCTL3	Bits [9:8]
MIPI DSI (LINK)	SYS_SLVACCCTL3	Bits [11:10]
LCD Controller	SYS_SLVACCCTL3	Bits [13:12]
USBPHY Control	SYS_SLVACCCTL3	Bits [17:16]
USB2.0 (channel 0)	SYS_SLVACCCTL3	Bits [19:18]
USB2.0 (channel 1)	SYS_SLVACCCTL3	Bits [21:20]
SDHI/eMMC (channel 0)	SYS_SLVACCCTL3	Bits [23:22]
SHDI (channel 1)	SYS_SLVACCCTL3	Bits [25:24]
GbEthernet (channel 0)	SYS_SLVACCCTL3	Bits [27:26]
GbEthernet (channel 1)	SYS_SLVACCCTL3	Bits [29:28]
I2C (channel 0)	SYS_SLVACCCTL4	Bits [1:0]
I2C (channel 1)	SYS_SLVACCCTL4	Bits [3:2]
I2C (channel 2)	SYS_SLVACCCTL4	Bits [5:4]
I2C (channel 3)	SYS_SLVACCCTL4	Bits [7:6]
CANFD	SYS_SLVACCCTL4	Bits [9:8]
RSPI	SYS_SLVACCCTL4	Bits [11:10]
SCIF (channel 0)	SYS_SLVACCCTL4	Bits [17:16]
SCIF (channel 1)	SYS_SLVACCCTL4	Bits [19:18]
SCIF (channel 2)	SYS_SLVACCCTL4	Bits [21:20]
SCIF (channel 3)	SYS_SLVACCCTL4	Bits [23:22]
SCIF (channel 4)	SYS_SLVACCCTL4	Bits [25:24]
SCI (channel 0)	SYS_SLVACCCTL4	Bits [27:26]
SCI (channel 1)	SYS_SLVACCCTL4	Bits [29:28]
SCI (channel 0) (IrDA)	SYS_SLVACCCTL4	Bits [31:30]
SSIF	SYS_SLVACCCTL5	Bits [1:0]
SRC	SYS_SLVACCCTL5	Bits [5:4]
ADC	SYS_SLVACCCTL6	Bits [1:0]
TSU	SYS_SLVACCCTL6	Bits [3:2]
OTP	SYS_SLVACCCTL7	Bits [3:2]
Registers within the SYSC to control the Cortex-M33*4	SYS_SLVACCCTL8	Bits [1:0]
Registers within the SYSC to control the Cortex-A55*4	SYS_SLVACCCTL8	Bits [3:2]
Register within the SYSC to control the LSI*4	SYS_SLVACCCTL10	Bits [1:0]
Registers within the SYSC to control the AOF*4	SYS_SLVACCCTL12	Bits [1:0]
Registers within the SYSC to control the Low Power Mode*4	SYS_SLVACCCTL13	Bits [1:0]
General-purpose registers within the SYSC*4	SYS_SLVACCCTL14	Bits [1:0]

Note 1. The security levels for the following bus slaves incorporated in this product are determined by using TZC-400. For details on the registers for controlling the security levels of these bus slaves, see **Table 5.8**.
— DDR3L and DDR4
— Memory areas of the SRAM ACU and SRAM MCPU
— Memory area, write buffer area, and control register area of SPI Multi I/O

Note 2. See **Section 6.3, Register Descriptions** for details on the registers.

Note 3. See **Section 6.3, Register Descriptions** for details on the bits.

Note 4. The SYSC has registers to control these functions. Setting of the security levels for each of the functions is possible. For details on the SYSC registers, see **Section 6.3, Register Descriptions**.

Table 5.8 Bus Slaves for which Security Levels are Determined by Using the Region ID Access Registers of the TZC

Target Bus Slave for Control	Security Control Unit (locations of the control registers)	Control Register*1	Control Bits*2
DDR	TZC (DDR)	REGION_ID_ACCESS_<n> (n: 0,1,2)*3	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
SPI Multi I/O	TZC (SPI Multi I/O)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
SRAM MCPU	TZC (SRAM MCPU)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]
SRAM ACPU	TZC (SRAM ACPU)	REGION_ID_ACCESS_0	nsaid_wr_en[31:16], nsaid_rd_en[15:0]

Note 1. For details on the related registers, see the Technical Reference Manual of CoreLink TZC-400 TrustZone Address Space Controller published by Arm Ltd.

Note 2. For details on the related bits, see the Technical Reference Manual of CoreLink TZC-400 TrustZone Address Space Controller published by Arm Ltd.

Note 3. REGION_ID_ACCESS_0 is for setting the security level for access to the DDR memory by using port 0 of the MEMC.
REGION_ID_ACCESS_1 is for setting the security level for access to the DDR memory by using port 1 of the MEMC.
REGION_ID_ACCESS_2 is for setting the security level for access to the DDR memory by using port 2 of the MEMC.
See **Section 13.1.2, Block Diagram** and **Figure 13.1** for the port number of the MEMC for use in access to the DDR memory from the bus masters.

5.4.2 Address Translation

5.4.2.1 34-Bit Address Space Access

This LSI has bus master units that can handle up to a 34-bit address space and bus master units that can only handle up to a 32-bit address space (actual size of 4 Gbytes). This function enables a bus master unit that can only handle up to a 32-bit address space (actual size of 4 Gbytes) to access an address space of greater than 4 Gbytes. The target bus master units are as follows.

- SDHI/eMMC
- GEthernet
- USB2.0
- DMAC
- FCPVD
- DSI-LINK

This function translates bus transaction addresses that are output from a bus master unit in 1-Gbyte units according to the settings of the address offset registers (SYS_AOFn: n = 0 to 6), as shown in the figure below. For details on the SYS_AOFn register, see **Section 6.3, Register Descriptions** in **Section 6, System Controller (SYSC)**.

- In the SYS_AOF registers, set bits [33:30] of the remapping destination address in each 1-Gbyte space corresponding to bits [31:30] of the address that is output from the bus master unit.
- Select the 4-bit value of the SYS_AOF register corresponding to the upper two bits (bits [31:30]) of the 32-bit address that is output from the bus master unit.
- In the selected 4-bit value, use the lower two bits to overwrite bits [31:30] of the original address that was output from the bus master unit, and add the upper two bits as bits [33:32] to the original address.

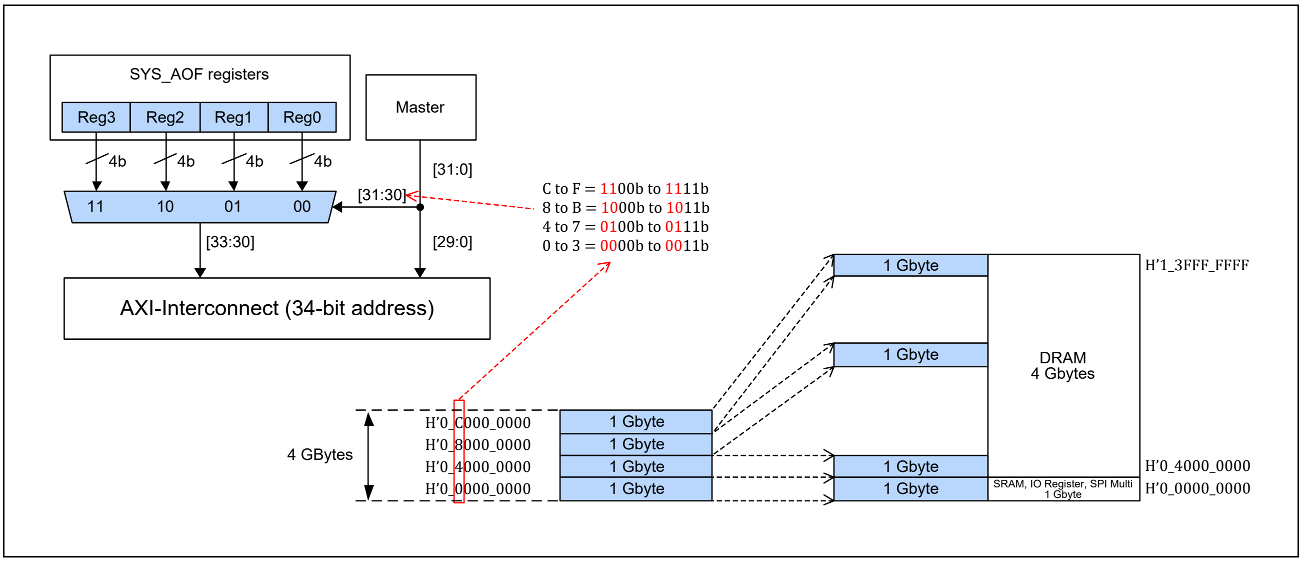


Figure 5.5 Address Space Extension (Overview)

CAUTION

When accessing DRAM from these bus master units, accesses that exceed a 4-Gbyte boundary are prohibited.

5.4.3 Bus Error Interrupt Generation

This LSI generates a bus error interrupt when any of the following conditions is met.

- A bus error is generated when access is disabled by the security setting.
- A bus error is generated when access to a slave unit in the module stop state (MSTOP) is attempted.
- A bus error is generated by a slave unit.

This function is supported for all bus master units included in this LSI.

Bus error interrupts are controlled (retained, cleared, and gathered into a single signal) by the interrupt synchronization circuits (IA55 and IM33). For details, including the types of the bus errors, see **Section 8, Interrupt Controller**.

NOTE

This function cannot distinguish between a bus error generated when access is disabled by the security setting and a bus error generated due to other causes.

In addition to the above bus error, an ERRINT interrupt is generated in any of the following cases.

- When the response to a write transaction with the bufferable attribute from the Cortex-M33 Code AHB interface is ERROR.
- When the response to a write transaction with the bufferable attribute from the Cortex-M33 System AHB interface is ERROR.

6. System Controller (SYSC)

6.1 Overview

6.1.1 Features

SYSC is a unit that performs system control of this LSI and has the following functions.

- Product information, external terminal state capture function
  - Management of product information of this LSI, and external terminal status

- Security control function
  - Re-setting the security attribute of transaction from each bus master
  - Judgement the security level of each bus slave (access management)

- 34-bit address space access function (see **Section 5.4.2.1, 34-Bit Address Space Access** for details)
  - If the bus master address is 32bit, do a 2bit extension to access the 34bit address space

- Low power consumption control (see **Section 42, Low Power Mode** for details)
  - Control the transition of this LSI to each mode of low power consumption
  - Setting cancellation conditions and confirming factors at the time of cancellation

- WDT stop control
  - Control WDT stop from CoreSight
  - Stop control of specified channel of WDT from CPU

6.1.2 Block Diagram of SYSC

The block diagram of SYSC is shown in **Figure 6.1**.

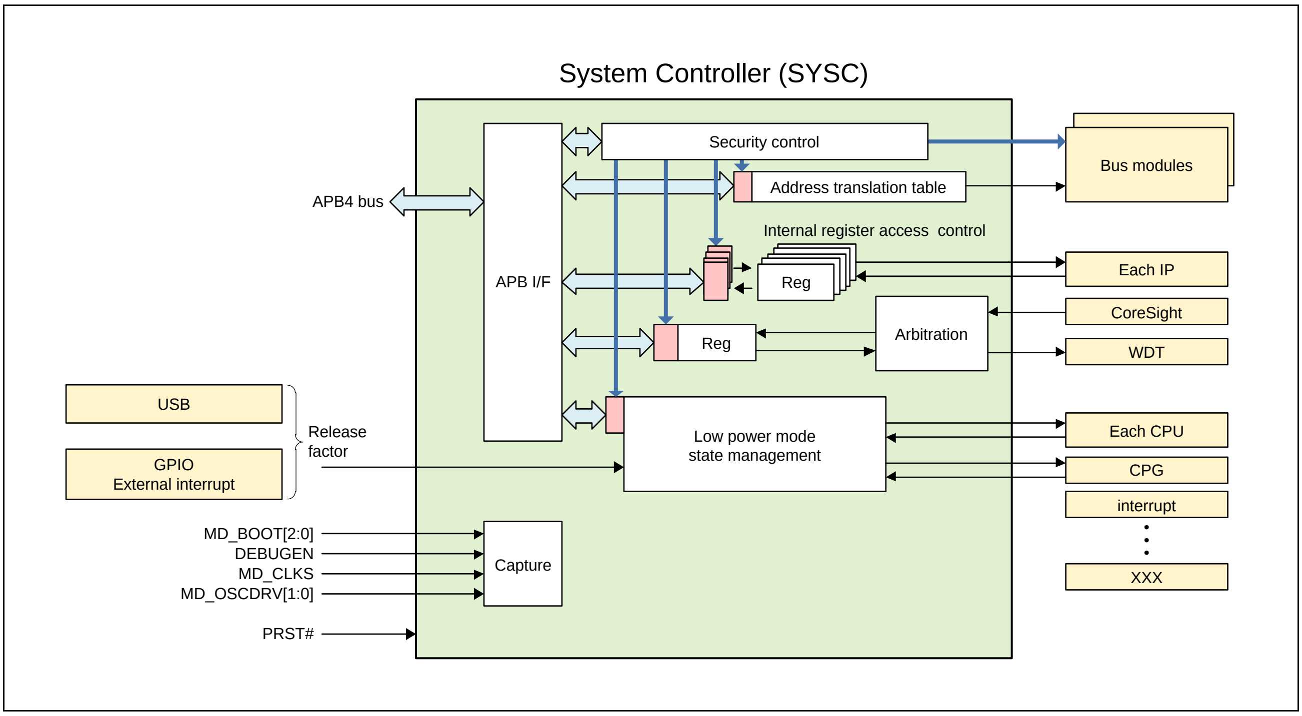


Figure 6.1 SYSC Block Diagram

6.2 Register Configuration

Table 6.1 shows the register configuration of SYSC. The address of the SYSC register is represented by the offset address from the base address. SYSC base address is as follows:

SYSC base address: H’0_1102_0000 (Overall Address Space)
SYSC base address: H’5102_0000 (Cortex-M33 Address Space Secure)
SYSC base address: H’4102_0000 (Cortex-M33 Address Space Non-Secure)

Table 6.1 SYSC Register Configuration (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Master Access Control Register0	SYS_MSTACCCTL0	R/W	H’00AA_AA00	H’0000	32
Master Access Control Register1	SYS_MSTACCCTL1	R/W	H’AAAA_AAAA	H’0004	32
Master Access Control Register2	SYS_MSTACCCTL2	R/W	H’00AA_AAAA	H’0008	32
Master Access Control Register3	SYS_MSTACCCTL3	R/W	H’00AA_AAAA	H’000C	32
Master Access Control Register4	SYS_MSTACCCTL4	R/W	H’AAAA_00AA	H’0010	32
Slave Access Control Register0	SYS_SLVACCCTL0	R/W	H’0AAA_AAA0	H’0100	32
Slave Access Control Register1	SYS_SLVACCCTL1	R/W	H’0800_C0AA	H’0104	32
Slave Access Control Register2	SYS_SLVACCCTL2	R/W	H’0000_0002	H’0108	32
Slave Access Control Register3	SYS_SLVACCCTL3	R/W	H’0000_0000	H’010C	32
Slave Access Control Register4	SYS_SLVACCCTL4	R/W	H’0000_0000	H’0110	32
Slave Access Control Register5	SYS_SLVACCCTL5	R/W	H’0000_0000	H’0114	32
Slave Access Control Register6	SYS_SLVACCCTL6	R/W	H’0000_0000	H’0114	32
Slave Access Control Register7	SYS_SLVACCCTL7	R/W	H’0000_0000	H’011C	32
Slave Access Control Register8	SYS_SLVACCCTL8	R/W	H’0000_0000	H’0120	32
Slave Access Control Register10	SYS_SLVACCCTL10	R/W	H’0000_0000	H’0128	32
Slave Access Control Register12	SYS_SLVACCCTL12	R/W	H’0000_0000	H’0130	32
Slave Access Control Register13	SYS_SLVACCCTL13	R/W	H’0000_0000	H’0134	32
Slave Access Control Register14	SYS_SLVACCCTL14	R/W	H’0000_0000	H’0138	32
ECCRAM0 ECC Setting Register	SYS_RAM0_ECC	R/W	H’0000_0000	H’0200	32
ECCRAM0 Access Control Register	SYS_RAM0_EN	R/W	H’0000_0003	H’0204	32
ECCRAM1 ECC Setting Register	SYS_RAM1_ECC	R/W	H’0000_0000	H’0210	32
ECCRAM1 Access Control Register	SYS_RAM1_EN	R/W	H’0000_0003	H’0214	32
WDT0 Control Register	SYS_WDT0_CTRL	R/W	H’0001_0000	H’0220	32
WDT1 Control Register	SYS_WDT1_CTRL	R/W	H’0001_0000	H’0230	32
WDT2 Control Register	SYS_WDT2_CTRL	R/W	H’0001_0000	H’0240	32
GEther0 Config Register	SYS_GETH0_CFG	R	H’0000_0000	H’0330	32
GEther1 Config Register	SYS_GETH1_CFG	R	H’0000_0000	H’0340	32
I2C0 Config Register	SYS_I2C0_CFG	R/W	H’0000_0000	H’0400	32
I2C1 Config Register	SYS_I2C1_CFG	R/W	H’0000_0000	H’0410	32
I2C2 Config Register	SYS_I2C2_CFG	R/W	H’0000_0000	H’0420	32
I2C3 Config Register	SYS_I2C3_CFG	R/W	H’0000_0000	H’0430	32
CM33 Config Register0	SYS_CM33_CFG0	R/W	H’0000_3D08	H’0804	32
CM33 Config Register1	SYS_CM33_CFG1	R/W	H’0000_3D08	H’0808	32
CM33 Config Register2	SYS_CM33_CFG2	R/W	H’1001_0000	H’080C	32
CM33 Config Register3	SYS_CM33_CFG3	R/W	H’0001_8000	H’0810	32
CM33 Lock Register	SYS_CM33_LOCK	R/W	H’0000_0000	H’0814	32

Table 6.1 SYSC Register Configuration (2/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
CA55 Core0 Reset Vector Address Low Configuration Register	SYS_CA55_CFG_RVAL0	R/W	H’0000_0000	H’0858	32
CA55 Core0 Reset Vector Address High Configuration Register	SYS_CA55_CFG_RVAH0	R/W	H’0000_0000	H’085C	32
CA55 Core1 Reset Vector Address Low Configuration Register	SYS_CA55_CFG_RVAL1	R/W	H’0002_0000	H’0860	32
CA55 Core1 Reset Vector Address High Configuration Register	SYS_CA55_CFG_RVAH1	R/W	H’0000_0000	H’0864	32
LSI Mode Signal Register	SYS_LSI_MODE	R	H’000x_xx0x	H’0A00	32
LSI Device ID Register	SYS_LSI_DEVID	R	H’x841_C447	H’0A04	32
LSI Product Register	SYS_LSI_PRR	R	H’0000_0xxx	H’0A08	32
Address Offset Register0	SYS_AOF0	R/W	H’3210_3210	H’0C00	32
Address Offset Register1	SYS_AOF1	R/W	H’3210_3210	H’0C04	32
Address Offset Register2	SYS_AOF2	R/W	H’3210_3210	H’0C08	32
Address Offset Register3	SYS_AOF3	R/W	H’0000_3210	H’0C0C	32
Address Offset Register4	SYS_AOF4	R/W	H’3210_3210	H’0C10	32
Address Offset Register5	SYS_AOF5	R/W	H’0000_3210	H’0C14	32
Address Offset Register6	SYS_AOF6	R/W	H’3210_3210	H’0C18	32
Lowpower Sequence Control Register1	SYS_LP_CTL1	R/W	H’0000_0000	H’0D04	32
Lowpower Sequence Control Register2	SYS_LP_CTL2	R/W	H’0000_0000	H’0D08	32
Lowpower Sequence Control Register5	SYS_LP_CTL5	R/W	H’0000_0000	H’0D14	32
Lowpower Sequence Control Register6	SYS_LP_CTL6	R/W	H’0000_0000	H’0D18	32
Lowpower Sequence Control Register7	SYS_LP_CTL7	R/W	H’0000_0000	H’0D1C	32
Lowpower Sequence CM33 Control Register0	SYS_LP_CM33CTL0	R/W	H’0x0x_0xxx	H’0D24	32
CA55 Clock Control Register1	SYS_LP_CA55CK_CTL1	R	H’0000_0x0x	H’0D38	32
CA55 Clock Control Register2	SYS_LP_CA55CK_CTL2	R/W	H’0000_0F06	H’0D3C	32
CA55 Clock Control Register3	SYS_LP_CA55CK_CTL3	R	H’0x0x_0x0x	H’0D40	32
GPU Lowpower Sequence Control Register	SYS_LP_GPU_CTL	R/W	H’xxxx_1Fxx	H’0D50	32
General Register0	SYS_GPREG_0	R/W	H’0000_0000	H’0E00	32
General Register1	SYS_GPREG_1	R/W	H’0000_0000	H’0E04	32
General Register2	SYS_GPREG_2	R/W	H’0000_0000	H’0E08	32
General Register3	SYS_GPREG_3	R/W	H’0000_0000	H’0E0C	32

6.3 Register Descriptions

6.3.1 Master Access Control Register 0 (SYS_MSTACCCTL0)

This register sets the secure attribute and privilege attribute of transactions from 3DGE and DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GPU_ARSEL	—	GPU_ARNS	GPU_ARRU	GPU_AWSEL	—	GPU_AWNS	GPU_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMAC1_ARSEL	—	DMAC1_ARNS	DMAC1_ARRU	DMAC1_AWSEL	—	DMAC1_AWNS	DMAC1_AWPU	DMAC0_ARSEL	—	DMAC0_ARNS	DMAC0_ARRU	DMAC0_AWSEL	—	DMAC0_AWNS	DMAC0_AWPU
Initial Value	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	GPU_ARSEL	0	R/W	3DGE read access security attribute source selection*1 0: ARPROT[1:0] is always 10b 1: GPU_ARPU is selected for ARPROT [0], and GPU_ARNS is selected for ARPROT [1].
22	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	GPU_ARNS	1	R/W	3DGE read access secure attribute*1 0: Secure 1: Non-secure
20	GPU_ARPU	0	R/W	3DGE read access privilege attribute*1 0: Non-privileged access 1: Privileged access
19	GPU_AWSEL	1	R/W	3DGE write access security attribute source selection*1 0: AWPROT[1:0] is always 10b 1: GPU_AWPU is selected for AWPROT [0], and GPU_AWNS is selected for AWPROT [1]
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	GPU_AWNS	1	R/W	3DGE write access secure attribute*1 0: Secure 1: Non-secure
16	GPU_AWPU	0	R/W	3DGE write access privilege attribute*1 0: Non-privileged access 1: Privileged access
15	DMAC1_ARSEL	1	R	Non-Secure DMAC read access secure attribute source selection The value of this register is always 1. The read value is always 1. For access from Non-Secure DMAC, ARPROT [1: 0] is always 10b.
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
13	DMAC1_ARNS	1	R	Non-Secure DMAC read access secure attribute 1: Non- Secure The value of this bit is always 1. The written value is ignored.
12	DMAC1_ARPU	0	R	Non-Secure DMAC read access privilege attribute 0: Non-privileged access The value of this bit is always 0. The written value is ignored.
11	DMAC1_AWSEL	1	R	Non-Secure DMAC write access security attribute source selection The value of this register is always 1. The read value is always 1 For access from Non-Secure DMAC, AWPROT [1: 0] is always 10b.
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	DMAC1_AWNS	1	R	Non-Secure DMAC write access secure attribute 1:Non- Secure The value of this bit is always 1. The written value is ignored.
8	DMAC1_AWPU	0	R	Non-Secure DMAC write access privilege attribute 0: Non-privileged access The value of this bit is always 0. The written value is ignored.
7	DMAC0_ARSEL	0	R/W	Secure DMAC read access security attribute source selection*1 0: ARPROT[1: 0] selects the value from Secure DMAC 1: DMAC0_ARPU is selected for ARPROT [0], and DMAC0_ARNS is selected for ARPROT [1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	DMAC0_ARNS	0	R/W	Secure DMAC read access secure attribute*1 0: Secure 1: Non-secure
4	DMAC0_ARPU	0	R/W	Secure DMAC read access privilege attribute*1 0: Non-privileged access 1: Privileged access
3	DMAC0_AWSEL	0	R/W	Secure DMAC write access security attribute source selection*1 0: AWPROT[1: 0] selects the value from Secure DMAC 1: DMAC0_AWPU is selected for AWPROT [0], and DMAC0_AWNS is selected for AWPROT [1].
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DMAC0_AWNS	0	R/W	Secure DMAC write access secure attribute*1 0: Secure 1: Non-secure
0	DMAC0_AWPU	0	R/W	Secure DMAC write access privilege attribute*1 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.2 Master Access Control Register 1 (SYS_MSTACCCTL1)

This register sets the secure attribute and privilege attribute of transactions from GbEthernet and SDHI/eMMC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GEther1_ARSEL	—	GEther1_ARNS	GEther1_ARRU	GEther1_AWSEL	—	GEther1_AWNS	GEther1_AWPU	GEther0_ARSEL	—	GEther0_ARNS	GEther0_ARRU	GEther0_AWSEL	—	GEther0_AWNS	GEther0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDHI1_ARSEL	—	SDHI1_ARNS	SDHI1_ARRU	SDHI1_AWSEL	—	SDHI1_AWNS	SDHI1_AWPU	SDHI0_ARSEL	—	SDHI0_ARNS	SDHI0_ARRU	SDHI0_AWSEL	—	SDHI0_AWNS	SDHI0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GEther1_ARSEL	1	R/W	GbEthernet (channel 1) read access security attribute source selection*[1] 0: ARPROT[1:0] is always 10b 1: GEther1_ARPU is selected for ARPROT [0], and GEther1_ARNS is selected for ARPROT[1].
30	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	GEther1_ARNS	1	R/W	GbEthernet (channel 1) read access secure attribute*[1] 0: Secure 1: Non-secure
28	GEther1_ARPU	0	R/W	GbEthernet (channel 1) read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
27	GEther1_AWSEL	1	R/W	GbEthernet (channel 1) write access security attribute source selection*[1] 0: AWPROT[1:0] is always 10b 1: GEther1_AWPU is selected for AWPROT [0], and GEther1_AWNS is selected for AWPROT[1]
26	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	GEther1_AWNS	1	R/W	GbEthernet (channel 1) write access secure attribute*[1] 0: Secure 1: Non-secure
24	GEther1_AWPU	0	R/W	GbEthernet (channel 1) write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
23	GEther0_ARSEL	1	R/W	GbEthernet (channel 0) read access security attribute source selection*[1] 0: ARPROT[1:0] is always 10b 1: GEther0_ARPU is selected for ARPROT [0], and GEther0_ARNS is selected for ARPROT[1].
22	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	GEther0_ARNS	1	R/W	GbEthernet (channel 0) read access secure attribute*[1] 0: Secure 1: Non-secure
20	GEther0_ARPU	0	R/W	GbEthernet (channel 0) read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access

Bit	Bit Name	Initial Value	R/W	Description
19	GEther0_AWSEL	1	R/W	GbEthernet (channel 0) write access security attribute source selection*[1] 0: AWPROT[1:0] is always 10b 1: GEther0_AWPU is selected for AWPROT [0], and GEther0_AWNS is selected for AWPROT [1].
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	GEther0_AWNS	1	R/W	GbEthernet (channel 0) write access secure attribute*[1] 0: Secure 1: Non-secure
16	GEther0_AWPU	0	R/W	GbEthernet (channel 0) weite access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
15	SDHI1_ARSEL	1	R/W	SDHI (channel 1) read access security attributed source selection*[1] 0: ARPROT[1:0] is always 10b 1: SDHI1_ARPU is selected for ARPROT [0], and SDHI1_ARNS is selected for ARPROT [1].
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	SDHI1_ARNS	1	R/W	SDHI (channel 1) read access secure attribute*[1] 0: Secure 1: Non-secure
12	SDHI1_ARPU	0	R/W	SDHI (channel 1) read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
11	SDHI1_AWSEL	1	R/W	SDHI (channel 1) write access security attribute source selection*[1] 0: AWPROT[1:0] is always 10b 1: SDHI1_AWPU is selected for AWPROT [0], and SDHI1_AWNS is selected for AWPROT [1].
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	SDHI1_AWNS	1	R/W	SDHI (channel 1) write access secure attribute*[1] 0: Secure 1: Non-secure
8	SDHI1_AWPU	0	R/W	SDHI (channel 1) write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
7	SDHI0_ARSEL	1	R/W	SDHI/eMMC (channel 0) read access security attributed source selection *[1] 0: ARPROT[1:0] is always 10b 1: SDHI0_ARPU is selected for ARPROT [0], and SDHI0_ARNS is selected for ARPROT [1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	SDHI0_ARNS	1	R/W	SDHI/eMMC (channel 0) read access secure attribute*[1] 0: Secure 1: Non-secure
4	SDHI0_ARPU	0	R/W	SDHI/eMMC (channel 0) read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
3	SDHI0_AWSEL	1	R/W	SDHI/eMMC (channel 0) write access security attribute source selection*[1] 0: AWPROT[1:0] is always 10b 1: SDHI0_AWPU is selected for AWPROT [0], and SDHI0_AWNS is selected for AWPROT [1].

Bit	Bit Name	Initial Value	R/W	Description
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	SDHI0_AWNS	1	R/W	SDHI/eMMC (channel 0) write access secure attribute*[1] 0: Secure 1: Non-secure
0	SDHI0_AWPU	0	R/W	SDHI/eMMC (channel 0) write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.3 Master Access Control Register 2 (SYS_MSTACCCTL2)

This register sets the secure attribute and privilege attribute of transactions from USB2.0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	USB2_0D_ARSEL	—	USB2_0D_ARNS	USB2_0D_ARRU	USB2_0D_AWSEL	—	USB2_0D_AWNS	USB2_0D_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	USB2_1H_ARSEL	—	USB2_1H_ARNS	USB2_1H_ARRU	USB2_1H_AWSEL	—	USB2_1H_AWNS	USB2_1H_AWPU	USB2_0H_ARSEL	—	USB2_0H_ARNS	USB2_0H_ARRU	USB2_0H_AWSEL	—	USB2_0H_AWNS	USB2_0H_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	USB2_0D_ARSEL	1	R/W	USB2.0 (channel 0, function) read access security attribute source selection[*1] 0: ARPROT[0] selects the HPROT[1] value from USB2.0 channel 0 Function, and ARPROT[1] is always 1. 1: USB2_0D_ARPU is selected for ARPROT[0], and USB2_0D_ARNS is selected for ARPROT[1].
22	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	USB2_0D_ARNS	1	R/W	USB2.0 (channel 0, function) read access secure attribute[*1] 0: Secure 1: Non-secure
20	USB2_0D_ARPU	0	R/W	USB2.0 (channel 0, function) read access privilege attribute[*1] 0: Non-privileged access 1: Privileged access
19	USB2_0D_AWSEL	1	R/W	USB2.0 (channel 0, function) write access security attribute source selection[*1] 0: AWPROT[0] selects the HPROT[1] value from USB2.0 channel 0 Function, and AWPROT [1] is always 1 1: USB2_0D_AWPU is selected for AWPROT[0], and USB2_0D_AWNS is selected for AWPROT[1].
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	USB2_0D_AWNS	1	R/W	USB2.0 (channel 0, function) write access secure attribute[*1] 0: Secure 1: Non-secure
16	USB2_0D_AWPU	0	R/W	USB2.0 (channel 0, function) write access privilege attribute[*1] 0: Non-privileged access 1: Privileged access
15	USB2_1H_ARSEL	0	R/W	USB2.0 (channel 1, host) read access security attribute source selection[*1] 0: ARPROT[0] selects the HPROT[1] value from USB2.0 channel 1 Host, and ARPROT[1] is always 1. 1: USB2_1H_ARPU is selected for ARPROT[0], and USB2_1H_ARNS is selected for ARPROT[1].
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
13	USB2_1H_ARNS	1	R/W	USB2.0 (channel 1, host) read access secure attribute*[1] 0: Secure 1: Non-secure
12	USB2_1H_ARPU	0	R/W	USB2.0 (channel 1, host) read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
11	USB2_1H_AWSEL	1	R/W	USB2.0 (channel 1, host) write access security attribute source selection*[1] 0: AWPROT[0] selects the HPROT[1] value from USB2.0 channel 1 Host, and AWPROT[1] is always 1. 1: USB2_1H_AWPU is selected for AWPROT[0], and USB2_1H_AWNS is selected for AWPROT[1].
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	USB2_1H_AWNS	1	R/W	USB2.0 (channel 1, host) write access secure attribute*[1] 0: Secure 1: Non-secure
8	USB2_1H_AWPU	0	R/W	USB2.0 (channel 1, host) write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
7	USB2_0H_ARSEL	0	R/W	USB2.0 (channel 0, host) read access security attribute source selection*[1] 0: ARPROT[0] selects the HPROT[1] value from USB2.0 channel 0 Host, and ARPROT[1] is always 1 1: USB2_0H_ARPU is selected for ARPROT[0], and USB2_0H_ARNS is selected for ARPROT[1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	USB2_0H_ARNS	1	R/W	USB2.0 (channel 0, host) read access secure attribute*[1] 0: Secure 1: Non-secure
4	USB2_0H_ARPU	0	R/W	USB2.0 (channel 0, host) read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
3	USB2_0H_AWSEL	1	R/W	USB2.0 (channel 0, host) write access security attribute source selection*[1] 0: AWPROT[1] selects the HPROT[1] value from USB2.0 channel 0 Host, and AWPROT[0] is always 1. 1: USB2_0H_AWPU is selected for AWPROT[0], and USB2_0H_AWNS is selected for AWPROT[1].
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	USB2_0H_AWNS	1	R/W	USB2.0 (channel 0, host) write access secure attribute*[1] 0: Secure 1: Non-secure
0	USB2_0H_AWPU	0	R/W	USB2.0 (channel 0, host) write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.4 Master Access Control Register 3 (SYS_MSTACCCTL3)

This register sets the secure attribute and privilege attribute of transactions from MIPI DSI, LCD controller and VCPL4.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DSI_ARSEL	—	DSI_ARNS	DSI_ARRU	DSI_AWSEL	—	DSI_AWNS	DSI_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCDC_ARSEL	—	LCDC_ARNS	LCDC_ARRU	LCDC_AWSEL	—	LCDC_AWNS	LCDC_AWPU	H264_ARSEL	—	H264_ARNS	H264_ARRU	H264_AWSEL	—	H264_AWNS	H264_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	DSI_ARSEL	1	R/W	MIPI DSI read access security attribute source selection*[1] 0: ARPROT[1:0] is always 10b 1: DSI_ARPU is selected for ARPROT[0], and DSI_ARNS is selected for ARPROT[1].
22	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	DSI_ARNS	1	R/W	MIPI DSI read access secure attribute*[1] 0: Secure 1: Non-secure
20	DSI_ARPU	0	R/W	MIPI DSI read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
19	DSI_AWSEL	1	R/W	MIPI DSI write access security attribute source selection*[1] 0: AWPROT[1:0] is always 10b 1: DSI_AWPU is selected for AWPROT[0], and DSI_AWNS is selected for AWPROT[1].
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	DSI_AWNS	1	R/W	MIPI DSI write access secure attribute*[1] 0: Secure 1: Non-secure
16	DSI_AWPU	0	R/W	MIPI DSI write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
15	LCDC_ARSEL	0	R/W	LCD Controller read access security attribute source selection*[1] 0: ARPROT[1:0] is always 10b 1: LCDC_ARPU is selected for ARPROT[0], and LCDC_ARNS is selected for ARPROT[1].
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	LCDC_ARNS	1	R/W	LCD Controller read access secure attribute*[1] 0: Secure 1: Non-secure

Bit	Bit Name	Initial Value	R/W	Description
12	LCDC_ARPU	0	R/W	LCD Controller read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
11	LCDC_AWSEL	1	R/W	LCD Controller write access security attribute source selection*[1] 0: AWPROT[1:0] is always 10b 1: LCDC_AWPU is selected for AWPROT[0], and LCDC_AWNS is selected for AWPROT[1].
10	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	LCDC_AWNS	1	R/W	LCD Controller write access secure attribute*[1] 0: Secure 1: Non-secure
8	LCDC_AWPU	0	R/W	LCD Controller write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
7	H264_ARSEL	0	R/W	VCPL4 read access security attribute source selection*[1] 0: ARPROT[1:0] is always 10b 1: H264_ARPU is selected for ARPROT[0], and H264_ARNS is selected for ARPROT[1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	H264_ARNS	1	R/W	VCPL4 read access secure attribute*[1] 0: Secure 1: Non-secure
4	H264_ARPU	0	R/W	VCPL4 read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
3	H264_AWSEL	1	R/W	VCPL4 write access security attribute source selection*[1] 0: AWPROT[1:0] is always 10b 1: H264_AWPU is selected for AWPROT[0], and H264_AWNS is selected for AWPROT[1].
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	H264_AWNS	1	R/W	VCPL4 write access secure attribute*[1] 0: Secure 1: Non-secure
0	H264_AWPU	0	R/W	VCPL4 write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.5 Master Access Control Register 4 (SYS_MSTACCCTL4)

This register sets the secure attribute and privilege attribute of transactions from CRU and ISU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	CRU_ST_AWSEL	—	CRU_ST_AWNS	CRU_ST_AWPU	—	—	—	—	CRU_VD_AWSEL	—	CRU_VD_AWNS	CRU_VD_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R/W	R	R/W	R/W	R	R	R	R	R/W	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ISU_ARSEL	—	ISU_ARNS	ISU_ARRU	ISU_AWSEL	—	ISU_AWNS	ISU_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'A	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
27	CRU_ST_AWSEL	1	R/W	CRU (Statistics) write access security attribute source selection*[1] Always write the initial value of 1 for this register. Operation is not guaranteed when 0 is written. 0: Setting prohibited 1: CRU_ST_AWPU is selected for AWPROT[0], and CRU_ST_AWNS is selected for AWPROT[1].
26	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	CRU_ST_AWNS	1	R/W	CRU (Statistics)write access secure attribute*[1] 0: Secure 1: Non-secure
24	CRU_ST_AWPU	0	R/W	CRU (Statistics) weite access privilege attribute*[1] 0: Non-privileged access 1: Priviledged access
23 to 20	—	H'A	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19	CRU_VD_AWSEL	1	R/W	CRU (Video) write access security attribute source selection*[1] Always write the initial value of 1 for this register. Operation is not guaranteed when 0 is written. 0: Setting prohibited 1: CRU_VD_AWPU is selected for AWPROT[0], and CRU_VD_AWNS is selected for AWPROT[1].
18	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CRU_VD_AWNS	1	R/W	CRU (Video)write access secure attribute*[1] 0: Secure 1: Non-secure
16	CRU_VD_AWPU	0	R/W	CRU (Video) weite access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	ISU_ARSEL	1	R/W	ISU read access security attributed source selection*[1] Always write the initial value of 1 for this register. Operation is not guaranteed when 0 is written. 0: Setting prohibited 1: ISU_ARPU is selected for ARPROT[0], and ISU_ARNS is selected for ARPROT[1].
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	ISU_ARNS	1	R/W	ISU read access secure attribute*[1] 0: Secure 1: Non-secure
4	ISU_ARPU	0	R/W	ISU read access privilege attribute*[1] 0: Non-privileged access 1: Privileged access
3	ISU_AWSEL	1	R/W	ISU write access security attribute source selection*[1] Always write the initial value of 1 for this register. Operation is not guaranteed when 0 is written. 0: Setting prohibited 1: ISU_AWPU is selected for AWPROT[0], and ISU_AWNS is selected for AWPROT[1].
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	ISU_AWNS	1	R/W	ISU write access secure attribute*[1] 0: Secure 1: Non-secure
0	ISU_AWPU	0	R/W	ISU write access privilege attribute*[1] 0: Non-privileged access 1: Privileged access

Note 1. This register setting is prohibited changing the setting while the target master is operating. Change it before the master initiates access.

6.3.6 Slave Access Control Register 0 (SYS_SLVACCCTL0)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the SRAM ACPU (Reg) and SRAM MCPU (Reg).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SRAM1_SL[1:0]		SRAM0_SL[1:0]	
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	H’0AAAAAA	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3, 2	SRAM1_SL[1:0]	00b	R/W	SRAM MCPU <Reg> slave access permission control Set the security level SL [1:0] for the slave SRAM MCPU <Reg>.
1, 0	SRAM0_SL[1:0]	00b	R/W	SRAM ACPU <Reg> slave access permission control Set the security level SL [1:0] for the slave SRAM ACPU <Reg>.

6.3.7 Slave Access Control Register 1 (SYS_SLVACCCTL1)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the DMAC, MHU, GPIO, IA55/IM33, GIC, SYC, SYSC, CPG, CoreSight and TZC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DMAC1_SL		DMAC0_SL		MHU_SL		GPIO_SL		IA55_IM33_SL		GIC_SL		SYS_SL	
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SYSC_SL		CPG_SL		CST_SL		—	—	TZC3_SL		TZC2_SL		TZC1_SL		TZC0_SL	
Initial Value	1	1	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29, 28	DMAC1_SL [1:0]	00b	R/W	Non-Secure DNAC slave access permission control Set the security level SL [1:0] for the slave Non-Secure DMAC.
27, 26	DMAC0_SL [1:0]	10b	R/W	Secure DMAC slave access permission control Set the security level SL [1:0] for the slave Secure DMAC.
25, 24	MHU_SL[1:0]	00b	R/W	MHU slave access permission control Set the security level SL [1:0] for the slave MHU.
23, 22	GPIO_SL[1:0]	00b	R/W	GPIO slave access permission control Set the security level SL [1:0] for the slave GPIO.
21, 20	IA55_IM33_SL [1:0]	00b	R/W	IA55/IM33 slave access permission control Set the security level SL [1:0] for the slave IA55/IM33.
19, 18	GIC_SL[1:0]	00b	R/W	GIC slave access permission control Set the security level SL [1:0] for the slave GIC.
17, 16	SYS_SL[1:0]	00b	R/W	SYC slave access permission control Set the security level SL [1:0] for the slave SYC.
15, 14	SYSC_SL[1:0]	11b	R	SYSC slave access permission control Reading the bits always returns 11b. Value cannot be written to these bits.
13, 12	CPG_SL[1:0]	00b	R/W	CPG slave access permission control Set the security level SL [1:0] for the slave CPG.
11, 10	CST_SL[1:0]	00b	R/W	CoreSight slave access permission control Set the security level SL [1:0] for the slave CoreSight.
9, 8	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7, 6	TZC3_SL[1:0]	10b	R/W	TZC (DDR) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (DDR) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.
5, 4	TZC2_SL[1:0]	10b	R/W	TZC (SPI Multi I/O) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (SPI Multi I/O) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.

Bit	Bit Name	Initial Value	R/W	Description
3, 2	TZC1_SL[1:0]	10b	R/W	TZC (SRAM MCPU) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (SRAM MCPU) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.
1, 0	TZC0_SL[1:0]	10b	R/W	TZC (SRAM ACPU) <Reg> slave access permission control Set the security level SL [1:0] for the slave TZC (SRAM ACPU) <Reg>. These bits can only be set to 10b or 11b. Setting them to 00b or 01b is prohibited. Operation is not guaranteed when 00b or 01b is set.

6.3.8 Slave Access Control Register 2 (SYS_SLVACCCTL2)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the GTM, WDT, MTU3a, POE3, GPT, POEG, and DDR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DDR_SL		POEG_SL		GPT_SL		POE3_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MTU3A_SL		—	—	WDT2_SL		WDT1_SL		WDT0_SL		OSTM2_SL		OSTM1_SL		OSTM0_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23, 22	DDR_SL	00b	R/W	DDR <Reg> slave access permission control Set the security level SL [1:0] for the slave DDR <Reg>.
21, 20	POEG_SL	00b	R/W	POEG slave access permission control Set the security level SL [1:0] for the slave POEG. 4 pins has a common security level.
19, 18	GPT_SL	00b	R/W	GPT slave access permission control Set the security level SL [1:0] for the slave GPT. 8 channels has a common security level.
17, 16	POE3_SL	00b	R/W	POE3 slave access permission control Set the security level SL [1:0] for the slave POE3.
15, 14	MTU3A_SL	00b	R/W	MTU3a slave access permission control Set the security level SL [1:0] for the slave MTU3a.
13, 12	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11, 10	WDT2_SL	00b	R/W	WDT channel 2 slave access permission control Set the security level SL [1:0] for the slave WDT channel 2.
9, 8	WDT1_SL	00b	R/W	WDT channel 1 slave access permission control Set the security level SL [1:0] for the slave WDT channel 1.
7, 6	WDT0_SL	00b	R/W	WDT channel 0 slave access permission control Set the security level SL [1:0] for the slave WDT channel 0.
5, 4	OSTM2_SL	00b	R/W	GTM channel 2 slave access permission control Set the security level SL [1:0] for the slave GTM channel 2.
3, 2	OSTM1_SL	00b	R/W	GTM channel 1 slave access permission control Set the security level SL [1:0] for the slave GTM channel 1.
1, 0	OSTM0_SL	10b	R/W	GTM channel 0 slave access permission control Set the security level SL [1:0] for the slave GTM channel 0.

6.3.9 Slave Access Control Register 3 (SYS_SLVACCCTL3)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the 3DGE, VCPL4, CRU, ISU, DSI, LCD Controller, USB2.0, SDHI/eMMC and GbEthernet.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ETH1_SL		ETH0_SL		SDHI1_SL		SDHI0_SL		USB21_SL		USB20_SL		USBT_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	LCDC_SL		DSILINK_SL		DSIPHY_SL		ISU_SL		CRU_SL		H264_SL		GPU_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29, 28	ETH1_SL	00b	R/W	GbEthernet channel1 slave access permission control Set the security level SL [1:0] for the slave GbEthernet channel 1.
27, 26	ETH0_SL	00b	R/W	GbEthernet channel 0 slave access permission control Set the security level SL [1:0] for the slave GbEthernet channel 0.
25, 24	SDHI1_SL	00b	R/W	SDHI channel 1 slave access permission control Set the security level SL [1:0] for the slave SDHI channel 1.
23, 22	SDHI0_SL	00b	R/W	SDHI/eMMC channel 0 slave access permission control Set the security level SL [1:0] for the slave SDHI/eMMC channel 0.
21, 20	USB21_SL	00b	R/W	USB2.0 channel 1 slave access permission control Set the security level SL [1:0] for the slave USB2.0 channel 1.
19, 18	USB20_SL	00b	R/W	USB2.0 channel 0 slave access permission control Set the security level SL [1:0] for the slave USB2.0 channel 0. USB 2.0 channel 0 has a common security level for Host and Function.
17, 16	USBT_SL	00b	R/W	USBPHY Control slave access permission control Set the security level SL [1:0] for the slave USBPHY Control.
15, 14	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13, 12	LCDC_SL	00b	R/W	LCD Controller slave access permission control Set the security level SL [1:0] for the slave LCD Controller.
11, 10	DSILINK_SL	00b	R/W	DSI (LINK) slave access permission control Set the security level SL [1:0] for the slave DSI (LINK).
9, 8	DSIPHY_SL	00b	R/W	DSI (PHY) slave access permission control Set the security level SL [1:0] for the slave DSI (PHY).
7, 6	ISU_SL	00b	R/W	ISU slave access permission control Set the security level SL [1:0] for the slave ISU.
5, 4	CRU_SL	00b	R/W	CRU slave access permission control Set the security level SL [1:0] for the slave CRU.
3, 2	H264_SL	00b	R/W	VCPL4 slave access permission control Set the security level SL [1:0] for the slave VCPL4.
1, 0	GPU_SL	00b	R/W	3DGE slave access permission control Set the security level SL [1:0] for the slave 3DGE.

6.3.10 Slave Access Control Register 4 (SYS_SLVACCCTL4)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the I2C, CANFD, RSPI, SCIF, SCI, and IrDA.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRDA_SL		SCI1_SL		SCI0_SL		SCIF4_SL		SCIF3_SL		SCIF2_SL		SCIF1_SL		SCIF0_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RSPI_SL		CANFD_SL		I2C3_SL		I2C2_SL		I2C1_SL		I2C0_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	IRDA_SL	00b	R	SCI channel 0 (IrDA) slave access permission control Set the security level SL [1:0] for the slave SCI channel 0 (IrDA).
29, 28	SCI1_SL	00b	R/W	SCI channel 1 slave access permission control Set the security level SL [1:0] for the slave SCI channel 1.
27, 26	SCI0_SL	00b	R/W	SCI channel 0 slave access permission control Set the security level SL [1:0] for the slave SCI channel 0.
25, 24	SCIF4_SL	00b	R/W	SCIF channel 4 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 4.
23, 22	SCIF3_SL	00b	R/W	SCIF channel 3 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 3.
21, 20	SCIF2_SL	00b	R/W	SCIF channel 2 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 2.
19, 18	SCIF1_SL	00b	R/W	SCIF channel 1 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 1.
17, 16	SCIF0_SL	00b	R/W	SCIF channel 0 slave access permission control Set the security level SL [1:0] for the slave SCIF channel 0.
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11, 10	RSPI_SL	00b	R/W	RSPI slave access permission control Set the security level SL [1:0] for the slave RSPI. RSPI has a common security level for channel 0 to channel 2.
9, 8	CANFD_SL	00b	R/W	CANFD slave access permission control Set the security level SL [1:0] for the slave CANFD.
7, 6	I2C3_SL	00b	R/W	I2C channel 3 slave access permission control Set the security level SL [1:0] for the slave I2C channel 3.
5, 4	I2C2_SL	00b	R/W	I2C channel 2 slave access permission control Set the security level SL [1:0] for the slave I2C channel 2.
3, 2	I2C1_SL	00b	R/W	I2C channel 1 slave access permission control Set the security level SL [1:0] for the slave I2C channel 1.
1, 0	I2C0_SL	00b	R/W	I2C channel 0 slave access permission control Set the security level SL [1:0] for the slave I2C channel 0.

6.3.11 Slave Access Control Register 5 (SYS_SLVACCCTL5)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the SSIF and SRC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SRC_SL		—	—	SSIF_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5, 4	SRC_SL	00b	R/W	SRC slave access permission control Set the security level SL [1:0] for the slave SRC.
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	SSIF_SL	00b	R/W	SSIF slave access permission control Set the security level SL [1:0] for the slave SSIF. SSIF has a common security level for channel 0 to channel 3.

6.3.12 Slave Access Control Register 6 (SYS_SLVACCCTL6)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the ADC and TSU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TSU_SL		ADC_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3, 2	TSU_SL	00b	R/W	TSU slave access permission control Set the security level SL [1:0] for the slave TSU.
1, 0	ADC_SL	00b	R/W	ADC slave access permission control Set the security level SL [1:0] for the slave ADC.

6.3.13 Slave Access Control Register 7 (SYS_SLVACCCTL7)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the OTP.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	OTP_SL		—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3, 2	OTP_SL	00b	R/W	OTP slave access permission control Set the security level SL [1:0] for the slave OTP.
1, 0	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.14 Slave Access Control Register 8 (SYS_SLVACCCTL8)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the Cortex-A55 control register and Cortex-M33 control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CA55_SL		CM33_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3, 2	CA55_SL	00b	R/W	SYSC <Cortex-A55 Control Reg> slave access permission control Set the security level SL [1:0] for the Cortex-A55 control register in SYSC.
1, 0	CM33_SL	00b	R/W	SYSC <Cortex-M33 Control Reg> slave access permission control Set the security level SL [1:0] for the Cortex-M33 control register in SYSC.

6.3.15 Slave Access Control Register 10 (SYS_SLVACCCTL10)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the LSI control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LSI_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	LSI_SL	00b	R/W	SYSC <LSI Control Reg> slave access permission control Set the security level SL [1:0] for the LSI control register in SYSC.

6.3.16 Slave Access Control Register 12 (SYS_SLVACCCTL12)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the AOF control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AOF_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	AOF_SL	00b	R/W	SYSC <AOF Control Reg> slave access permission control Set the security level SL [1:0] for the AOF control register in SYSC.

6.3.17 Slave Access Control Register 13 (SYS_SLVACCCTL13)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the Low Power Mode control register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LP_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	LP_SL	00b	R/W	SYSC <Low Power Mode Control Reg> slave access permission control Set the security level SL [1:0] for the Low Power Mode control register in SYSC.

6.3.18 Slave Access Control Register 14 (SYS_SLVACCCTL14)

This register sets the security levels for determining whether to enable or disable access to each set of registers in the general-purpose register in SYSC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GPREG_SL	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	GPREG_SL	00b	R/W	SYSC <GPREG> slave access permission control Set the security level SL [1:0] for the General purpose register(GPREG) in SYSC.

6.3.19 ECCRAM0 ECC Setting Register (SYS_RAM0_ECC)

This register is a register that enables and disables the ECC function of ECCRAM0 (SRAM ACPU).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VECCEN	0	R/W	Set enable or disable error correction for ECCRAM0 (SRAM ACPU) 0: ECC function disabled 1: ECC function enabled

6.3.20 ECCRAM0 Access Control Register (SYS_RAM0_EN)

This register is a register that controls access to ECCRAM0 (SRAM ACPU).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	VLWEN	1	R/W	Set enable or disable writes to ECRAM0 (SRAM ACPU) 0: Write disabled 1: Write enabled
0	VCEN	1	R/W	Set enable or disable access to ECRAM0 (SRAM ACPU) 0: Access disabled 1: Access enabled

6.3.21 ECCRAM1 ECC Setting Register (SYS_RAM1_ECC)

This register is a register that enables and disables the ECC function of ECCRAM1 (SRAM MCPU).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VECCEN	0	R/W	Set enable or disable error correction for ECCRAM1 (SRAM MCPU) 0: ECC function disabled 1: ECC function enabled

6.3.22 ECCRAM1 Access Control Register (SYS_RAM1_EN)

This register is the register that controls access to ECCRAM1 (SRAM M CPU).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	VLWEN	1	R/W	Set enable or disable writes to ECCRAM1 (SRAM MCPU) 0: Write disabled 1: Write enabled
0	VCEN	1	R/W	Set enable or disable access to ECRAM1 (SRAM MCPU) 0: Access disabled 1: Access enabled

6.3.23 WDT0 Control Register (SYS_WDT0_CTRL)

This register is the register that set the count control of WDT0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	WDTSTOPMASK	1	R/W	Controls the stop function of WDT0 (for Cortex-A55 Core0) from CoreSight during debugging. 0: Stops WDT0 counting during debugging. 1: Not stop WDT0 counting during debugging.
15 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	WDTSTOP	0	R/W	Set whether the WDT0 (for Cortex-A55 Core0) counter is forced to stop 0: Continue counting 1: Stop counting and hold the counter value

6.3.24 WDT1 Control Register (SYS_WDT1_CTRL)

This register is the register that set the count control of WDT1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	WDTSTOPMASK	1	R/W	Controls the stop function of WDT1 (for Cortex-A55 Core1) from CoreSight during debugging. 0: Stops WDT1 counting during debugging. 1: Not stop WDT1 counting during debugging.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	WDTSTOP	0	R/W	Set whether the WDT1 (for Cortex-A55 Core1) counter is forced to stop 0: Continue counting 1: Stop counting and hold the counter value

6.3.25 WDT2 Control Register (SYS_WDT2_CTRL)

This register is the register that set the count control of WDT2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	WDTSTOPMASK	1	R/W	Controls the stop function of WDT2 (for Cortex-M33) from CoreSight during debugging. 0: Stops WDT2 counting during debugging. 1: Not stop WDT2 counting during debugging.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	WDTSTOP	0	R/W	Set whether the WDT2 (for Cortex-M33) counter is forced to stop 0: Continue counting 1: Stop counting and hold the counter value

6.3.26 GEther0 Config Register (SYS_GETH0_CFG)

This register is the register that indicates the state of Ether ch0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FEC_GIGA_ENABLE	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	FEC_GIGA_ENABLE	0	R	Indicates the mode of speed for Ether ch0 0: 100M/1Mbps mod 1: 1Gbps mode
23 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.27 GEther1 Config Register (SYS_GETH1_CFG)

This register is the register that indicates the state of Ether ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FEC_GIGA_ENABLE	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	FEC_GIGA_ENABLE	0	R	Indicates the mode of speed for Ether ch1 0: 100M/1Mbps mode 1: 1Gbps mode
23 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.28 I2C0 Config Register (SYS_I2C0_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC0_SDA, RIIC0_SCL) of I2C ch0. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.29 I2C1 Config Register (SYS_I2C1_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC1_SDA, RIIC1_SCL) of I2C ch1. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.30 I2C2 Config Register (SYS_I2C2_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC2_SDA, RIIC2_SCL) of I2C ch2. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.31 I2C3 Config Register (SYS_I2C3_CFG)

This register is a register that set the bypass mode of the analog filter of I2C ch3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	af_bypass
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	af_bypass	0	R/W	Set whether to bypass the analog filter for noise removal for the input terminals (RIIC3_SDA, RIIC3_SCL) of I2C ch3. 0: Noise reduction is performed using an analog filter for the input signal. 1: The input signal is bypassed an analog filter and noise reduction is not performed.

6.3.32 CM33 Config Register0 (SYS_CM33_CFG0)

This register is the register that set the secure Systick of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CONFIGSSYSTICK[25:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CONFIGSSYSTICK[25:0]															
Initial Value	0	0	1	1	1	1	0	1	0	0	0	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 0	CONFIGSSYSTICK[25:0]	H'000_3D08	R/W	Configuration register for secure Systick. The value must be determined before the cold reset of Cortex-M33 is released. See the Cortex-M33 chapter for more information on settings.

6.3.33 CM33 Config Register1 (SYS_CM33_CFG1)

This register is the register that set cortex-M33 non-secure Systick.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CONFIGNSSYSTICK[25:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CONFIGNSSYSTICK[25:0]															
Initial Value	0	0	1	1	1	1	0	1	0	0	0	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 0	CONFIGNSSYSTICK[25:0]	H'000_3D08	R/W	Configuration register for non-secure Systick. The value must be determined before the cold reset of Cortex-M33 is released. See the Cortex-M33 chapter for more information on settings.

6.3.34 CM33 Config Register2 (SYS_CM33_CFG2)

This register is the register that set the secure vector address of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INITSVTOR[31:7]															
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INITSVTOR[31:7]									—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITSVTOR [31:7]	H'020_0200	R/W	A register that indicates the secure vector address. The address value must be determined before the Cortex-M33 cold reset is released.
6 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.35 CM33 Config Register3 (SYS_CM33_CFG3)

This register is the register that set the non-secure vector address of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INITNSVTOR[31:7]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INITNSVTOR[31:7]									—	—	—	—	—	—	—
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITNSVTOR [31:7]	H’000_0300	R/W	A register that indicates the non-secure vector address. The address value must be determined before the Cortex-M33 cold reset is released.
6 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.36 CM33 Lock Register (SYS_CM33_LOCK)

This register is a register that set the vector-address change permission of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LOCKNSVTOR	LOCKSVTAIRCR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	LOCKNSVTOR	0	R/W	Control the non-secure vector address change permission. 0: Changeable 1: Cannot be changed
0	LOCKSVTAIRCR	0	R/W	Control changing permission of the Secure Vector address, and the PRIS bit and the BFHFNMINS bit of the AIRCR register of Cortex-M33. 0: Changeable 1: Cannot be changed

6.3.37 CA55 Core0 Reset Vector Address Low Configuration Register (SYS_CA55_CFG_RVAL0)

This register is a register that set the reset vector address of Cortex-A55 core0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RVBARADDRL0[31:2]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RVBARADDRL0[31:2]														—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	RVBARADDRL0[31: 2]	H’0000_0000	R/W	Represent bottom bit 31 to bit 2 of the reset vector address of Cortex-A55 Core0. When you change a vector address, you must set the address value and then release the reset (Cold reset/Cluster Warm reset/Core Warm reset). See **Section 2, System CPU Cortex-A55** for the sequence of each reset.
1, 0	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.38 CA55 Core0 Reset Vector Address High Configuration Register (SYS_CA55_CFG_RVAH0)

This register is a register that set the reset vector address of Cortex-A55 core0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RVBARADDRH0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	RVBARADDRH0[7:0]	H’00	R/W	Represent upper bit 39 to bit 32 of the reset vector address of Cortex-A55 Core0. When you change a vector address, you must set the address value and then release the reset (Cold reset/Cluster Warm reset/Core Warm reset). See **Section 2, System CPU Cortex-A55** for the sequence of each reset.

6.3.39 CA55 Core1 Reset Vector Address Low Configuration Register (SYS_CA55_CFG_RVAL1)

This register is a register that set the reset vector address of Cortex-A55 core1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RVBARADDRL1[31:2]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RVBARADDRL1[31:2]														—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	RVBARADDRL1[31:2]	H'0000_8000	R/W	Represent bottom bit 31 to bit 2 of the reset vector address of Cortex-A55 Core1. When you change a vector address, you must set the address value and then release the reset (Cold reset/Cluster Warm reset/Core Warm reset). See **Section 2, System CPU Cortex-A55** for the sequence of each reset.
1, 0	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

6.3.40 CA55 Core1 Reset Vector Address High Configuration Register (SYS_CA55_CFG_RVAH1)

This register is a register that set the reset vector address of Cortex-A55 core1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RVBARADDRH1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	RVBARADDRH1[7:0]	H'00	R/W	Represent upper bit 39 to bit 32 of the reset vector address of Cortex-A55 Core1. When you change a vector address, you must set the address value and then release the reset (Cold reset/Cluster Warm reset/Core Warm reset). See **Section 2, System CPU Cortex-A55** for the sequence of each reset.

6.3.41 LSI Mode Signal Register (SYS_LSI_MODE)

This register indicates the state of the MD_BOOT terminal, the DEBUGEN terminal, the MD_CLKS terminal, and the MD_OSCDRV terminal.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT_SEC_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STAT_MD_OSCDRV[1:0]		—	STAT_MD_CLKS	—	—	STAT_DEBUGEN	—	—	—	—	—	—	STAT_MD_BOOT[2:0]		
Initial Value	—	—	0	—	0	0	—	0	0	0	0	0	0	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	STAT_SEC_EN	—	R	Indicates whether this LSI support the security function or not. 0: Not supported 1: Available
15, 14	STAT_MD_OSCDRV[1:0]	—	R	Indicates the terminal state of the external terminal MD_OSCDRV[1:0]. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.
13	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	STAT_MD_CLKS	—	R	Indicates the terminal state of the external terminal MD_CLKS. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.
11, 10	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	STAT_DEBUGEN	—	R	Indicates the terminal state of the external terminal DEBUGEN. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.
8	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	STAT_MD_BOOT[2:0]	—	R	Indicates the terminal state of the external terminal MD_BOOT[2:0]. The value is updated at the rising edge of PRST#. If a reset occurs inside the LSI, this register is not updated.

6.3.42 LSI Device ID Register (SYS_LSI_DEVID)

This register indicates the product specific device ID.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEV_ID[31:28]				DEV_ID[27:0]											
Initial Value	—	—	—	—	1	0	0	0	0	1	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEV_ID[27:0]															
Initial Value	1	1	0	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DEV_ID [31:28]	—	R	Indicates the product revision.
27 to 0	DEV_ID[27:0]	H’841_C447	R	Indicates the product specific fixed value. *Note*: RZ/G2L and RZ/G2LC are the same value. If identification between G2L and G2LC is needed, read the value of address H’0_1186_1178. The value is below. H’0000_0000: RZ/G2L H’0000_0001: RZ/G2LC

6.3.43 LSI Product Register (SYS_LSI_PRR)

This register indicates information about product options.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CA55_1 CPU
Initial Value	0	0	0	0	0	0	0	—	0	0	0	—	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CA55_1CPU	—	R	Indicates whether the CA55 is single or dual. 0: Cortex-A55 Dual 1: Cortex-A55 Single

6.3.44 Address Offset Register0 (SYS_AOF0)

This register is an address offset register for accessing the 34-bit address space from SD ch0/ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXSDHI_1				OFS10_SXSDHI_1				OFS01_SXSDHI_1				OFS00_SXSDHI_1			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXSDHI_0				OFS10_SXSDHI_0				OFS01_SXSDHI_0				OFS00_SXSDHI_0			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXSDHI_1	0011b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 11*1
27 to 24	OFS10_SXSDHI_1	0010b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 10*1
23 to 20	OFS01_SXSDHI_1	0001b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 01*1
19 to 16	OFS00_SXSDHI_1	0000b	R/W	Conversion value to address [33:30] when SD ch1 address [31:30] = 00*1
15 to 12	OFS11_SXSDHI_0	0011b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 11*1
11 to 8	OFS10_SXSDHI_0	0010b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 10*1
7 to 4	OFS01_SXSDHI_0	0001b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 01*1
3 to 0	OFS00_SXSDHI_0	0000b	R/W	Conversion value to address [33:30] when SD ch0 address [31:30] = 00*1

Note 1. Master SD ch0/ch1 are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.45 Address Offset Register1 (SYS_AOF1)

This register is an address offset register for accessing the 34-bit address space from Ether ch0/ch1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXGIGE_1				OFS10_SXGIGE_1				OFS01_SXGIGE_1				OFS00_SXGIGE_1			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXGIGE_0				OFS10_SXGIGE_0				OFS01_SXGIGE_0				OFS00_SXGIGE_0			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXGIGE_1	0011b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 11*[1]
27 to 24	OFS10_SXGIGE_1	0010b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 10*[1]
23 to 20	OFS01_SXGIGE_1	0001b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 01*[1]
19 to 16	OFS00_SXGIGE_1	0000b	R/W	Conversion value to address [33:30] when Ether ch1 address [31:30] = 00*[1]
15 to 12	OFS11_SXGIGE_0	0011b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 11*[1]
11 to 8	OFS10_SXGIGE_0	0010b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 10*[1]
7 to 4	OFS01_SXGIGE_0	0001b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 01*[1]
3 to 0	OFS00_SXGIGE_0	0000b	R/W	Conversion value to address [33:30] when Ether ch0 address [31:30] = 00*[1]

Note 1. Master Ether ch0/ch1 are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.46 Address Offset Register2 (SYS_AOF2)

This register is an address offset register for accessing the 34-bit address space from USB2.0 ch0/ch1 Host.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXUSB2_1				OFS10_SXUSB2_1				OFS01_SXUSB2_1				OFS00_SXUSB2_1			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXUSB2_0_H				OFS10_SXUSB2_0_H				OFS01_SXUSB2_0_H				OFS00_SXUSB2_0_H			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXUSB2_1	0011b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 11*1
27 to 24	OFS10_SXUSB2_1	0010b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 10*1
23 to 20	OFS01_SXUSB2_1	0001b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 01*1
19 to 16	OFS00_SXUSB2_1	0000b	R/W	Conversion value to address [33:30] when USB2.0 ch1 Host address [31:30] = 00*1
15 to 12	OFS11_SXUSB2_0_H	0011b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 11*1
11 to 8	OFS10_SXUSB2_0_H	0010b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 10*1
7 to 4	OFS01_SXUSB2_0_H	0001b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 01*1
3 to 0	OFS00_SXUSB2_0_H	0000b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Host address [31:30] = 00*1

Note 1. Master USB2.0 ch0/ch1 Host are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.47 Address Offset Register3 (SYS_AOF3)

This register is an address offset register for accessing the 34-bit address space from USB2.0 ch0 Function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXUSB2_0_F				OFS10_SXUSB2_0_F				OFS01_SXUSB2_0_F				OFS00_SXUSB2_0_F			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 12	OFS11_SXUSB2_0_F	0011b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 11*1
11 to 8	OFS10_SXUSB2_0_F	0010b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 10*1
7 to 4	OFS01_SXUSB2_0_F	0001b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 01*1
3 to 0	OFS00_SXUSB2_0_F	0000b	R/W	Conversion value to address [33:30] when USB2.0 ch0 Function address [31:30] = 00*1

Note 1. Master USB2.0 ch0 Function is the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.48 Address Offset Register4 (SYS_AOF4)

This register is an address offset register for accessing the 34-bit address space from LCDC and DSI.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXDSIL				OFS10_SXDSIL				OFS01_SXDSIL				OFS00_SXDSIL			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXLCDC				OFS10_SXLCDC				OFS01_SXLCDC				OFS00_SXLCDC			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXDSIL	0011b	R/W	Conversion value to address [33:30] when DSI address [31:30] = 11[*1]
27 to 24	OFS10_SXDSIL	0010b	R/W	Conversion value to address [33:30] when DSI address [31:30] = 10[*1]
23 to 20	OFS01_SXDSIL	0001b	R/W	Conversion value to address [33:30] when DSI address [31:30] = 01[*1]
19 to 16	OFS00_SXDSIL	0000b	R/W	Conversion value to address [33:30] when DSI address [31:30] = 00[*1]
15 to 12	OFS11_SXLCDC	0011b	R/W	Conversion value to address [33:30] when LCDC address [31:30] = 11[*1]
11 to 8	OFS10_SXLCDC	0010b	R/W	Conversion value to address [33:30] when LCDC address [31:30] = 10[*1]
7 to 4	OFS01_SXLCDC	0001b	R/W	Conversion value to address [33:30] when LCDC address [31:30] = 01[*1]
3 to 0	OFS00_SXLCDC	0000b	R/W	Conversion value to address [33:30] when LCDC address [31:30] = 00[*1]

Note 1. Master LCDC and DSI are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.49 Address Offset Register5 (SYS_AOF5)

This register is an address offset register for accessing the 34-bit address space from FCPCS.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXH264				OFS10_SXH264				OFS01_SXH264				OFS00_SXH264			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 12	OFS11_SXH264	0011b	R/W	Conversion value to address [33:30] when FCPCS address [31:30] = 11*1
11 to 8	OFS10_SXH264	0010b	R/W	Conversion value to address [33:30] when FCPCS address [31:30] = 10*1
7 to 4	OFS01_SXH264	0001b	R/W	Conversion value to address [33:30] when FCPCS address [31:30] = 01*1
3 to 0	OFS00_SXH264	0000b	R/W	Conversion value to address [33:30] when FCPCS address [31:30] = 00*1

Note 1. Master FCPCS is the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.50 Address Offset Register6 (SYS_AOF6)

This register is an address offset register for accessing the 34-bit address space from Secure/Non-Secure DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFS11_SXDMAC_NS				OFS10_SXDMAC_NS				OFS01_SXDMAC_NS				OFS00_SXDMAC_NS			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS11_SXDMAC_S				OFS10_SXDMAC_S				OFS01_SXDMAC_S				OFS00_SXDMAC_S			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	OFS11_SXDMAC_NS	0011b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 11*1
27 to 24	OFS10_SXDMAC_NS	0010b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 10*1
23 to 20	OFS01_SXDMAC_NS	0001b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 01*1
19 to 16	OFS00_SXDMAC_NS	0000b	R/W	Conversion value to address [33:30] when Non-Secure DMAC address [31:30] = 00*1
15 to 12	OFS11_SXDMAC_S	0011b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 11*1
11 to 8	OFS10_SXDMAC_S	0010b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 10*1
7 to 4	OFS01_SXDMAC_S	0001b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 01*1
3 to 0	OFS00_SXDMAC_S	0000b	R/W	Conversion value to address [33:30] when Secure DMAC address [31:30] = 00*1

Note 1. Master Secure/Non-Secure DMAC are the master of 32bit addresses (4 Gbytes space). Depending on the address value (in 1 GB) of the master address [31:30], you can access the address converted to the address space [33:30]. This gives you access to more than 4 Gbytes of space. This register setting is prohibited from changing the setting while the target master is in operation. Please change it while the access on the master side is stopped.

6.3.51 SYS_LP_CTL1

This register is a register that requests the transition to low power consumption mode and confirms the status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CM33SLEEP_ACK	—	—	CA55SLEEP_ACK		—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CM33SLEEP_REQ	—	—	CA55SLEEP_REQ		—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	CM33SLEEP_ACK	0b	R/W	Cortex-M33 Sleep Mode transition response 0: There is no response of transition to Cortex-M33 Sleep Mode 1: There is a response of transition to Cortex-M33 Sleep Mode
27, 26	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25, 24	CA55SLEEP_ACK	00b	R/W	Cortex-A55 Sleep Mode transition response 00: There is no response of transition to Cortex-A55 Sleep Mode 01: There is a response of transition to Cortex-A55 Core0 Sleep Mode 10: There is a response of transition to Cortex-A55 Core1 Sleep Mode 11: There is a response of transition to Cortex-A55 Core0 and core1 Sleep Mode
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
16	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	CM33SLEEP_REQ	0b	R/W	Cortex-M33 Sleep Mode transition request 0: There is no transition request to Cortex-M33 Sleep Mode 1: There is a transition request to Cortex-M33 Sleep Mode
11, 10	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	CA55SLEEP_REQ	00b	R/W	Cortex-A55 Sleep Mode transition request 00: There is no transition request to Cortex-A55 Sleep Mode 01: There is a transition request to Cortex-A55 Core0 Sleep Mode 10: There is a transition request to Cortex-A55 Core1 Sleep Mode 11: There is a transition request to Cortex-A55 Core0 and core1 Sleep Mode
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.52 SYS_LP_CTL2

This register is set before and after Cortex-A55 Sleep Mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CA55_STBYCTL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CA55_STBYCTL	0b	R/W	Registers to be set before and after cortex-A55 Sleep Mode. 0: Cortex-A55 Sleep Mode operation complete. When using in other modes, be sure to 0. 1: Cortex-A55 Sleep Mode start

6.3.53 SYS_LP_CTL5

This register controls the low power mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CM33SLEEP_F	CA55SLEEP1_F	CA55SLEEP0_F	—	—	—	—	—	AMCLKQDENY_F	ASCLKQDENY_F	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
19 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	CM33SLEEP_F	0b	R/W	Cortex-M33 Sleep Mode request flag 0: No Cortex-M33 Sleep Mode request (clear with 0 write) 1: Cortex-M33 Sleep Mode request (SYS_LPM_INT interrupt occurs)
9	CA55SLEEP1_F	0b	R/W	Cortex-A55 core1 Sleep Mode request flag 0: No Cortex-A55 core1 Sleep Mode request (clear with 0 write) 1: Cortex-A55 core1 Sleep Mode request (SYS_LPM_INT interrupt occurs)
8	CA55SLEEP0_F	0b	R/W	Cortex-A55 core0 Sleep Mode request flag 0: No Cortex-A55 core0 Sleep Mode request (clear with 0 write) 1: Cortex-A55 core0 Sleep Mode request (SYS_LPM_INT interrupt occurs)
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQDENY_F	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Master Interface QDENY Factor AMCLKQDENY Flag 0: No Deny response (clear with 0 write) 1: Deny response (SYS_CA55_DENY interrupt occurs)
1	ASCLKQDENY_F	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Slave Interface QDENY Factor ASCLKQDENY Flag 0: No Deny response (clear with 0 write) 1: Deny response (SYS_CA55_DENY interrupt occurs)

Bit	Bit Name	Initial Value	R/W	Description
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.54 SYS_LP_CTL6

This register controls the low power mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CM33SLEEP_E	CA55SLEEP1_E	CA55SLEEP0_E	—	—	—	—	—	AMCLKQDENY_E	ASCLKQDENY_E	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	CM33SLEEP_E	0b	R/W	Cortex-M33 Sleep Mode Requirements Mask 0: Mask Cortex-M33 Sleep Mode requirements factor 1: Don't mask Cortex-M33 Sleep Mode requirements factor
9	CA55SLEEP1_E	0b	R/W	Cortex-A55 core1 Sleep Mode Requirements Mask 0: Mask Cortex-A55 core1 Sleep Mode requirements factor 1: Don't mask Cortex-A55 core1 Sleep Mode requirements factor
8	CA55SLEEP0_E	0b	R/W	Cortex-A55 core0 Sleep Mode Requirements Mask 0: Mask Cortex-A55 core0 Sleep Mode requirements factor 1: Don't mask Cortex-A55 core0 Sleep Mode requirements factor
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQDENY_E	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Master Interface QDENY Configuration Factor Mask 0: Masks the configuration factors AMCLKQDENY_F for cortex-A55 ACE asynchronous bridge master interface QDENY 1: Do not mask the configuration factor AMCLKQDENY_F for Cortex-A55 ACE Asynchronous Bridge Master Interface QDENY
1	ASCLKQDENY_E	0b	R/W	Cortex-A55 ACE Asynchronous Bridge Slave Interface QDENY Configuration Factor Mask 0: Masks the configuration factors ASCLKQDENY_F for cortex-A55 ACE asynchronous bridge slave interface QDENY 1: Do not mask the configuration factor ASCLKQDENY_F for Cortex-A55 ACE Asynchronous bridge slave Interface QDENY

Bit	Bit Name	Initial Value	R/W	Description
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.55 SYS_LP_CTL7

This register controls the low power mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IM33_MASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	IM33_MASK	0b	R/W	IM33 to Cortex-M33 Interrupt Mask Settings Set this bit before entering Cortex-M33 Sleep Mode (Deep Sleep), Cortex-M33 Warm Reset. The interrupt mask setting is automatically removed by the hardware. 0: Do not mask interrupts from IM33 to Cortex-M33. 1: Mask interrupts from IM33 to Cortex-M33.

6.3.56 SYS_LP_CM33CTL0

This register controls the low power mode of Cortex-M33.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	—	0	0	0	0	0	0	0	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SYSRESETREQ	—	—	—	—	SLEEPDEEP	—	—	—	SLEEPMODE
Initial Value	0	0	0	0	0	0	—	—	0	0	0	—	0	0	0	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
24	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
16	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	SYSRESETREQ	—	R	Warm reset request from Cortex-M33 0: No request 1: Requested
8	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	SLEEPDEEP	—	R	Cortex-M33 SLEEPDEEP status 0: Cortex-M33 is not in the Sleep Mode 1: Cortex-M33 is either in the Sleep Mode For more information, see **Section 3, System CPU Cortex-M33**.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SLEEPMODE	—	R	Cortex-M33 Sleep mode status 0: Not in the Sleep Mode. 1: In the Sleep Mode Asserted in multiple modes. For more information, see **Section 3, System CPU Cortex-M33**.

6.3.57 SYS_LP_CA55CK_CTL1

This register controls the low power mode of Cortex-A55.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PDBGCLKQACTIVE	GICCLKQACTIVE	ATCLKQACTIVE	PCLKQACTIVE	—	—	—	—	—	AMCLKQACTIVE	ASCLKQACTIVE	—
Initial Value	0	0	0	0	—	—	—	—	0	0	0	0	0	—	—	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	PDBGCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the PDBGCLK domain of Cortex-A55 cluster.*1
10	GICCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the GICCLK domain of Cortex-A55 cluster.*1
9	ATCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the ATCLK domain of Cortex-A55 cluster.*1
8	PCLKQACTIVE	—	R	Represents the status of a QACTIVE signal from the PCLK domain of Cortex-A55 cluster.*1
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQACTIVE	—	R	Represents the state of the QACTIVE signal from cortex-A55 ACE asynchronous bridge Master Interface.*1
1	ASCLKQACTIVE	—	R	Represents the state of the QACTIVE signal from the Cortex-A55 ACE asynchronous bridge Slave Interface.*1
0	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.

Note 1. The state of this register is used to determine whether subsequent Q-channel controls will start clock stop operations. For more information about QACTIVE signal specifications, see AMBA® Low Power Interface ARM® Q-Channel and P-Channel Interfaces.

6.3.58 SYS_LP_CA55CK_CTL2

This register controls the low power mode of Cortex-A55.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PDBGCLKQREQn	GICCLKQREQn	ATCLKQREQn	PCLKQREQn	—	—	—	—	—	AMCLKQREQn	ASCLKQREQn	—
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	PDBGCLKQREQn	1b	R/W	Set the level of the QREQn signal to the PDBGCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with PDBGCLKQACCEPTn and PDBGCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
10	GICCLKQREQn	1b	R/W	Set the level of the QREQn signal to the GICCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with GICCLKQACCEPTn and GICCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
9	ATCLKQREQn	1b	R/W	Set the level of the QREQn signal to the ATCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with ATCLKQACCEPTn and ATCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
8	PCLKQREQn	1b	R/W	Set the level of the QREQn signal to the PCLK domain of the Cortex-A55 cluster. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with PCLKQACCEPTn and PCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQREQn	1b	R/W	Set the level of the QREQn signal to the ACE asynchronous bridge Master Interface on cortex-A55. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with AMCLKQACCEPTn and AMCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.

Bit	Bit Name	Initial Value	R/W	Description
1	ASCLKQREQn	1b	R/W	Set the level of the QREQn signal to the ACE asynchronous bridge Slave Interface on cortex-A55. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with ASCLKQACCEPTn and ASCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

6.3.59 SYS_LP_CA55CK_CTL3

This register controls the low power mode of Cortex-A55.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PDBGCLKQDENY	GICCLKQDENY	ATCLKQDENY	PCLKQDENY	—	—	—	—	—	AMCLKQDENY	ASCLKQDENY	CA55_COREINSTRRUN[1]
Initial Value	0	0	0	0	—	—	—	—	0	0	0	0	0	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PDBGCLKQACCEPTn	GICCLKQACCEPTn	ATCLKQACCEPTn	PCLKQACCEPTn	—	—	—	—	—	AMCLKQACCEPTn	ASCLKQACCEPTn	CA55_COREINSTRRUN[0]
Initial Value	0	0	0	0	—	—	—	—	0	0	0	0	0	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	0000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	PDBGCLKQDENY	—	R	Represents the status of the QDENY signal from the PDBGCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PDBGCLKQREQn and PDBGCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
26	GICCLKQDENY	—	R	Represents the status of the QDENY signal from the GICCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with GICCLKQREQn and GICCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
25	ATCLKQDENY	—	R	Represents the status of the QDENY signal from the ATCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with ATCLKQREQn and ATCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
24	PCLKQDENY	—	R	Represents the status of the QDENY signal from the PCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PCLKQREQn and PCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
23 to 19	—	00000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	AMCLKQDENY	—	R	Represents the status of the QDENY signal from the ACE asynchronous bridge Master Interface on cortex-A55. This register represents the state of the Q-channel in combination with AMCLKQREQn and AMCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.

Bit	Bit Name	Initial Value	R/W	Description
17	ASCLKQDENY	—	R	Represents the status of the QDENY signal from the ACE asynchronous bridge Slave Interface on cortex-A55. This register represents the state of the Q-channel in combination with ASCLKQREQn and ASCLKQACCEPTn. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
16	CA55_COREINSTRRUN[1]	—	R	Indicates that Cortex-A55 Core1 has transitioned to the WFI/WFE state. 0: Transitioned or OFF state 1: Normal operation
15 to 12	—	0000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	PDBGCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the PDBGCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PDBGCLKQREQn and PDBGCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
10	GICCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the GICCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with GICCLKQREQn and GICCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
9	ATCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the ATCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with ATCLKQREQn and ATCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
8	PCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the PCLK domain of a Cortex-A55 cluster. This register represents the state of the Q-channel in combination with PCLKQREQn and PCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
7 to 3	—	00000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	AMCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the ACE asynchronous bridge Master Interface on cortex-A55. This register represents the state of the Q-channel in combination with AMCLKQREQn and AMCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
1	ASCLKQACCEPTn	—	R	Represents the state of the QACCEPTn signal from the ACE asynchronous bridge Slave Interface on cortex-A55. This register represents the state of the Q-channel in combination with ASCLKQREQn and ASCLKQDENY. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
0	CA55_COREINSTRRUN[0]	—	R	Indicates that Cortex-A55 Core0 has transitioned to the WFI/WFE state. 0: Transitioned or OFF state 1: Normal operation

6.3.60 SYS_LP_GPU_CTL

This register controls the low power mode of GPU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	QDENY_ACE_MST	QDENY_ACE_SLV	QDENY_AXI_MST	QDENY_AXI_SLV	QDENY_GPU	—	—	—	QACCEPTn_ACE_MST	QACCEPTn_ACE_SLV	QACCEPTn_AXI_MST	QACCEPTn_AXI_SLV	QACCEPTn_GPU
Initial Value	0	0	0	—	—	—	—	—	0	0	0	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	QREQn_ACE_MST	QREQn_ACE_SLV	QREQn_AXI_MST	QREQn_AXI_SLV	QREQn_GPU	—	—	—	QACTIVE_ACE_MST	QACTIVE_ACE_SLV	QACTIVE_AXI_MST	QACTIVE_AXI_SLV	QACTIVE_GPU
Initial Value	0	0	0	1	1	1	1	1	0	0	0	—	—	—	—	—
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	QDENY_ACE_MST	—	R	ACE-lite MST Q-Channel request denial status Represents the state of the QDENYn signal from the GPU ACE-lite asynchronous bridge Master Interface. This register represents the state of the Q-channel in combination with QREQn_ACE_MST and QACCEPTn_ACE_MST. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
27	QDENY_ACE_SLV	—	R	ACE-lite SLV Q-Channel request denial status Represents the state of the QDENYn signal from the GPU ACE-lite asynchronous bridge Slave Interface. This register represents the state of the Q-channel in combination with QREQn_ACE_SLV and QACCEPTn_ACE_SLV. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
26	QDENY_AXI_MST	—	R	AXI4 MST Q-Channel request denial status Represents the state of the QDENYn signal from the GPU AXI4 asynchronous bridge Master Interface. This register represents the state of the Q-channel in combination with QREQn_AXI_MST and QACCEPTn_AXI_MST. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
25	QDENY_AXI_SLV	—	R	AXI4 SLV Q-Channel request denial status Represents the state of the QDENYn signal from the GPU AXI4 asynchronous bridge Slave Interface. This register represents the state of the Q-channel in combination with QREQn_AXI_SLV and QACCEPTn_AXI_SLV. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
24	QDENY_GPU	—	R	GPU Q-Channel request denial status Represents the state of the QDENYn signal from the GPU. This register represents the state of the Q-channel in combination with QREQn_GPU and QACCEPTn_GPU. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
23 to 21	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
20	QACCEPTn_ ACE_MST	—	R	ACE-lite MST Q-Channel request accept status Represents the state of the QACCEPTn signal from the GPU ACE-lite asynchronous bridge Master Interface. This register represents the state of the Q-channel in combination with QREQn_ACE_MST and QDENY_ACE_MST. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
19	QACCEPTn_ ACE_SLV	—	R	ACE-lite SLV Q-Channel request accept status Represents the state of the QACCEPTn signal from the GPU ACE-lite asynchronous bridge Slave Interface. This register represents the state of the Q-channel in combination with QREQn_ACE_SLV and QDENY_ACE_SLV. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
18	QACCEPTn_ AXI_MST	—	R	AXI4 MST Q-Channel request accept status Represents the state of the QACCEPTn signal from the GPU AXI4 asynchronous bridge Master Interface. This register represents the state of the Q-channel in combination with QREQn_AXI_MST and QDENY_AXI_MST. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
17	QACCEPTn_ AXI_SLV	—	R	AXI4 SLV Q-Channel request accept status Represents the state of the QACCEPTn signal from the GPU AXI4 asynchronous bridge Slave Interface. This register represents the state of the Q-channel in combination with QREQn_AXI_SLV and QDENY_AXI_SLV. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
16	QACCEPTn_ GPU	—	R	GPU Q-Channel request accept status Represents the state of the QACCEPTn signal from the GPU. This register represents the state of the Q-channel in combination with QREQn_GPU and QDENY_GPU. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	QREQn_ACE _MST	1b	R/W	ACE-lite MST Q-Channel request status Set the level of the QREQn signal to the GPU ACE-lite asynchronous bridge Master Interface. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with QACCEPTn_ACE_MST and QDENY_ACE_MST. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
11	QREQn_ACE _SLV	1b	R/W	ACE-lite SLV Q-Channel request status Set the level of the QREQn signal to the GPU ACE-lite asynchronous bridge Slave Interface. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with QACCEPTn_ACE_SLV and QDENY_ACE_SLV. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
10	QREQn_AXI_ MST	1b	R/W	AXI4 MST Q-Channel request status Set the level of the QREQn signal to the GPU AXI4 asynchronous bridge Master Interface. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with QACCEPTn_AXI_MST and QDENY_AXI_MST. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.

Bit	Bit Name	Initial Value	R/W	Description
9	QREQn_AXI_SLV	1b	R/W	AXI4 SLV Q-Channel request status Set the level of the QREQn signal to the GPU AXI4 asynchronous bridge Slave Interface. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with QACCEPTn_AXI_SLV and QDENY_AXI_SLV. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
8	QREQn_GPU	1b	R/W	GPU Q-Channel request status Set the level of the QREQn signal to the GPU. When this register is read, the value set at the time of the write is read. This register represents the state of the Q-channel in combination with QACCEPTn_GPU and QDENY_GPU. For more information on these combinations, see AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces.
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	QACTIVE_ACE_MST	—	R	ACE-lite MST Q-Channel active status Represents the state of the QACTIVE signal from the GPU ACE-lite asynchronous bridge Master Interface.*1
3	QACTIVE_ACE_SLV	—	R	ACE-lite SLV Q-Channel active status Represents the state of the QACTIVE signal from the GPU ACE-lite asynchronous bridge Slave Interface.*1
2	QACTIVE_AXI_MST	—	R	AXI4 MST Q-Channel active status Represents the state of the QACTIVE signal from the GPU AXI4 asynchronous bridge Master Interface.*1
1	QACTIVE_AXI_SLV	—	R	AXI4 SLV Q-Channel active status Represents the state of the QACTIVE signal from the GPU AXI4 asynchronous bridge Slave Interface.*1
0	QACTIVE_GPU	—	R	GPU Q-Channel active status Represents the state of the QACTIVE signal from the GPU.*1

Note 1. The state of this register is used to determine whether subsequent Q-channel controls will start clock stop operations. For more information about QACTIVE signal specifications, see AMBA® Low Power Interface ARM® Q-Channel and P-Channel Interfaces.

6.3.61 General Register0 (SYS_GPREG_0)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG0 [31:0]	H’0000_0000	R/W	General-purpose register 0 It is a 32-bit register. It is possible to write any value, read the written value.

6.3.62 General Register1 (SYS_GPREG_1)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG1 [31:0]	H’0000_0000	R/W	General-purpose register 1 It is a 32-bit register. It is possible to write any value, read the written value.

6.3.63 General Register2 (SYS_GPREG_2)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG2[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG2[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG2 [31:0]	H'0000_0000	R/W	General-purpose register 2 It is a 32-bit register. It is possible to write any value, read the written value.

6.3.64 General Register3 (SYS_GPREG_3)

This register is a general-purpose register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG3[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG3[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG3 [31:0]	H'0000_0000	R/W	General-purpose register 3 It is a 32-bit register. It is possible to write any value, read the written value.

6.4 Operation

6.4.1 External Terminal State Capture Function

Captures the state of the following four external terminals at the rising edge of PRST# terminal and holds them into the SYS_LSI_MODE register. The value is not updated by a reset that occurs inside the LSI.

- MD_BOOT[2:0]
- DEBUGEN
- MD_CLKS
- MD_OSCDRV[1:0]

6.4.2 WDT Stop Control Function

By writing 1 to the WDTSTOP bit of the SYS_WDTn_CTRL register (n: 0 to 2), the following WDT corresponding to n is forcibly stopped and the count value is retained.

Also, by writing 1 to the WDTSTOPMASK bit of the SYS_WDTn_CTRL register (n: 0 to 2), the following WDT count corresponding to n can be continued even during debugging. If 0 is written, the WDT count will stop during debugging.

n	Stop WDT
0	WDT0 (for Cortex-A55 core0)
1	WDT1 (for Cortex-A55 core1)
2	WDT2 (for Cortex-M33)

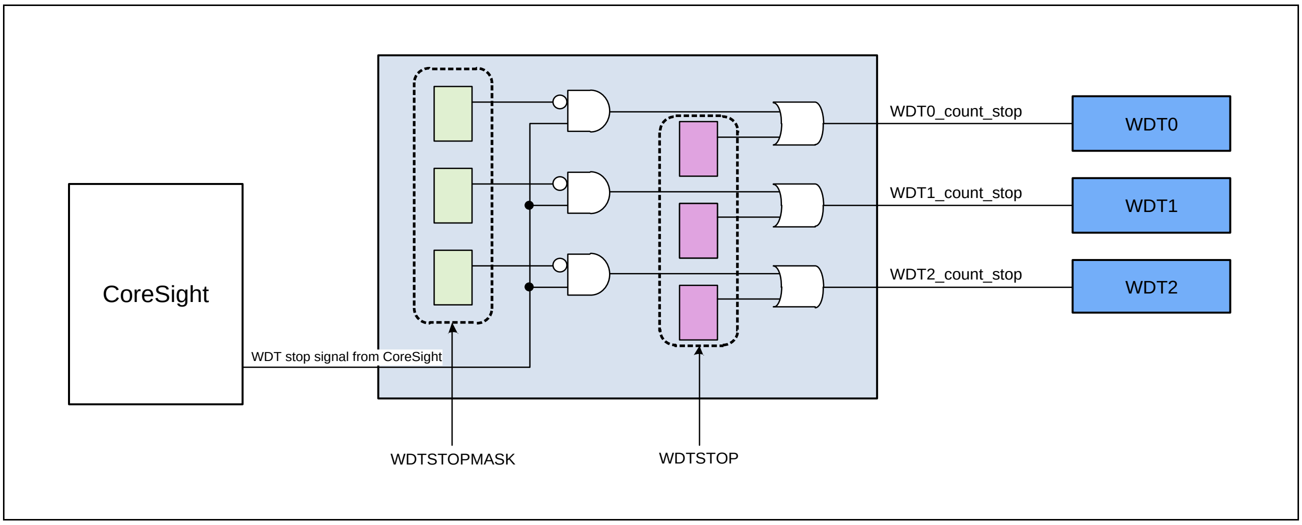


Figure 6.2 Overview of WDT Stop Function

7. Clock Pulse Generator (CPG)

This chapter describes the functions of the clock pulse generator module (CPG).

7.1 Features

7.1.1 Overview

Table 7.1 provides a list of CPG features.

Table 7.1 List of functions

Features	Specification
PLL control	● SSCG control, PLL multiplication factor setting, and ON/OFF control are performed by register setting. ● This LSI is equipped with 6 PLLs. The roles of each are as follows: – PLL1 (SSCG-PLL: Default SSCG ON): Cortex-A55 only – PLL2 (SSCG-PLL: Default SSCG OFF): For SYSTEM-BUS & Module (no SSCG) – PLL3 (SSCG-PLL: Default SSCG ON): For SYSTEM-BUS & Module (SSCG Yes) – PLL4 (SSCG-PLL: Default SSCG ON): DDR-SDRAM Only – PLL5 (SSCG-PLL: Default SSCG ON): Video-Out (DSI/LCDC) / Gigabit Ethernet Interface – PLL6 (SSCG-PLL: Default SSCG ON): 3DGE / Gigabit Ethernet Interface ● Initial value of SSCG for each PLL (PLL3 is fixed): – Fmod: 31.25kHz – Spread: Down spread – Modulation Depth: 2.25% (PLL1, PLL4 and, PLL6) 2.20% (PLL3 and PLL5)
Clock generation and control	The clock to be supplied to each module is generated from the external clock input (EXCLK), crystal oscillator input / output (XIN / XOUT), or PLL output clock. ● Selection of the peri-ratio by register setting ● Clock path selection by clock selector by register setting ● Clock supply ON/OFF control is performed by register setting. (Module Standby Mode control) ● Cortex-M33 Sleep Mode control
Reset generation and control	A reset is generated from the reset factor in (). The types of reset are as follows. ● System reset (external pin) ● Software reset (system reboot, reset control of each module) ● Watchdog reset (WDT module) ● Cortex-A55 Warm reset control (software control) ● Cortex-M33 Warm reset control (software control) ● ON/OFF control for each unit clock (software control)
Monitor	● PLL clock monitor – The function to monitor the operation of the output clocks from PLL1 to PLL6 ● Clock monitor for each module – Function to monitor that the clock monitor is operating ● Reset monitor – Function to monitor the reset status of each module

7.1.2 CPG configuration

Figure 7.1 shows the CPG configuration.

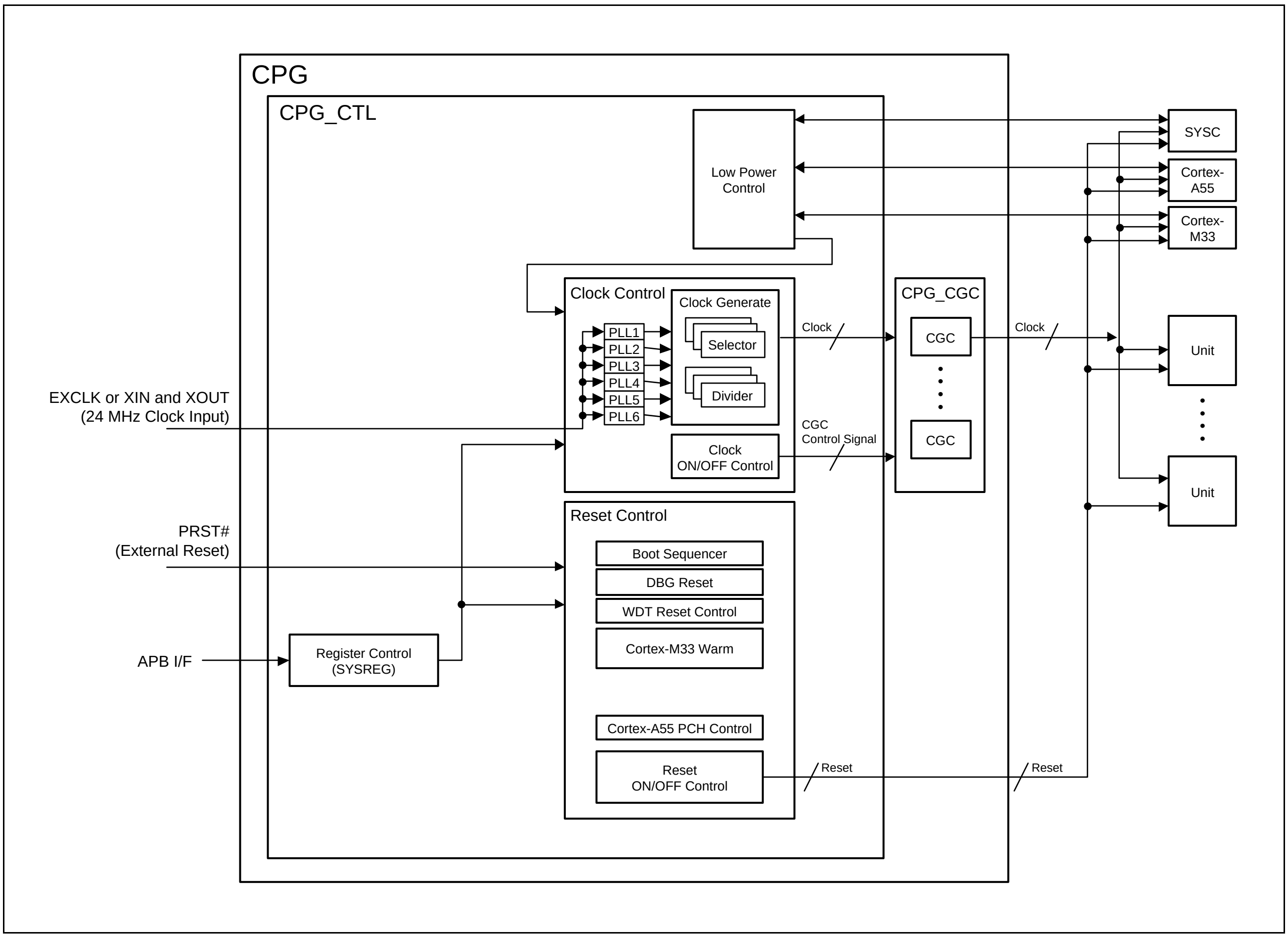


Figure 7.1 CPG Configuration

7.1.3 Pin description

The table below shows a list of CPG pins.

Table 7.2 Pin List

Name	Pin Name	IO	Functions
Mode control pin	MD_BOOT0	I	Set the operating mode. Do not change during PRST# pin asserts and after negating until the operating mode is confirmed.
	MD_BOOT1	I	
	MD_BOOT2	I	
	MD_CLKS	I	Set ON/OFF of SSCG circuit operation of PLL3. Do not change during PRST# terminal asserts and after negating until the operating mode is confirmed.
External clock Input pin*[2]	EXCLK	I	This is the pin to input the external clock.
Crystal oscillator input/output pin*[1]	XIN*[3]	I	This is the pin to connect the crystal oscillator.
	XOUT*[3]	O	
Power on reset input pin	PRST#	I	When this pin becomes low level, it will be in the power on reset state.
OSC drive capacity setting pin 0	MD_OSCDRV0	I	This is a pin for adjusting the drive capacity of the OSC buffer.
OSC drive capacity setting pin 1	MD_OSCDRV1	I	This is a pin for adjusting the drive capacity of the OSC buffer.

Note 1. When using a crystal oscillator, fix the pins of MD_OSCDRV0 and MD_OSCDRV1 to Low Level. Also, refer to the C/R constraints and implementation precautions described in the PCB design checklist.

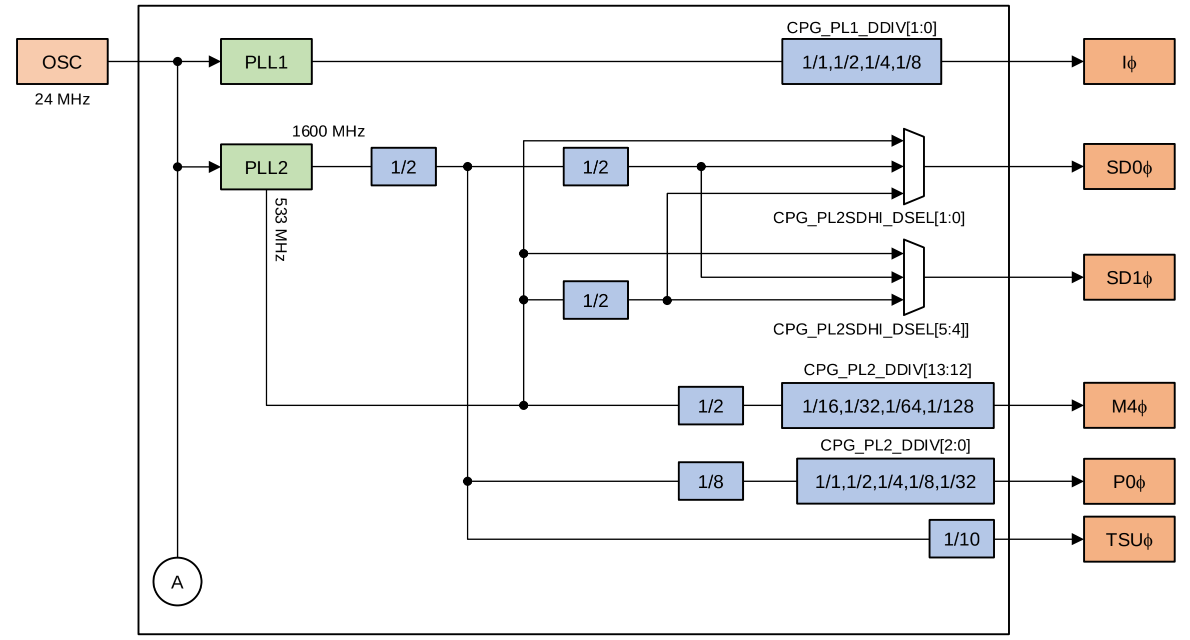
Note 2. When using crystal oscillation, fix the EXCLK pin externally to Low Level.

Note 3. When using the clock from the outside without using the crystal oscillator, set the XIN pin to the external low level fixing process and the XOUT pin to Open.

7.1.4 Clock

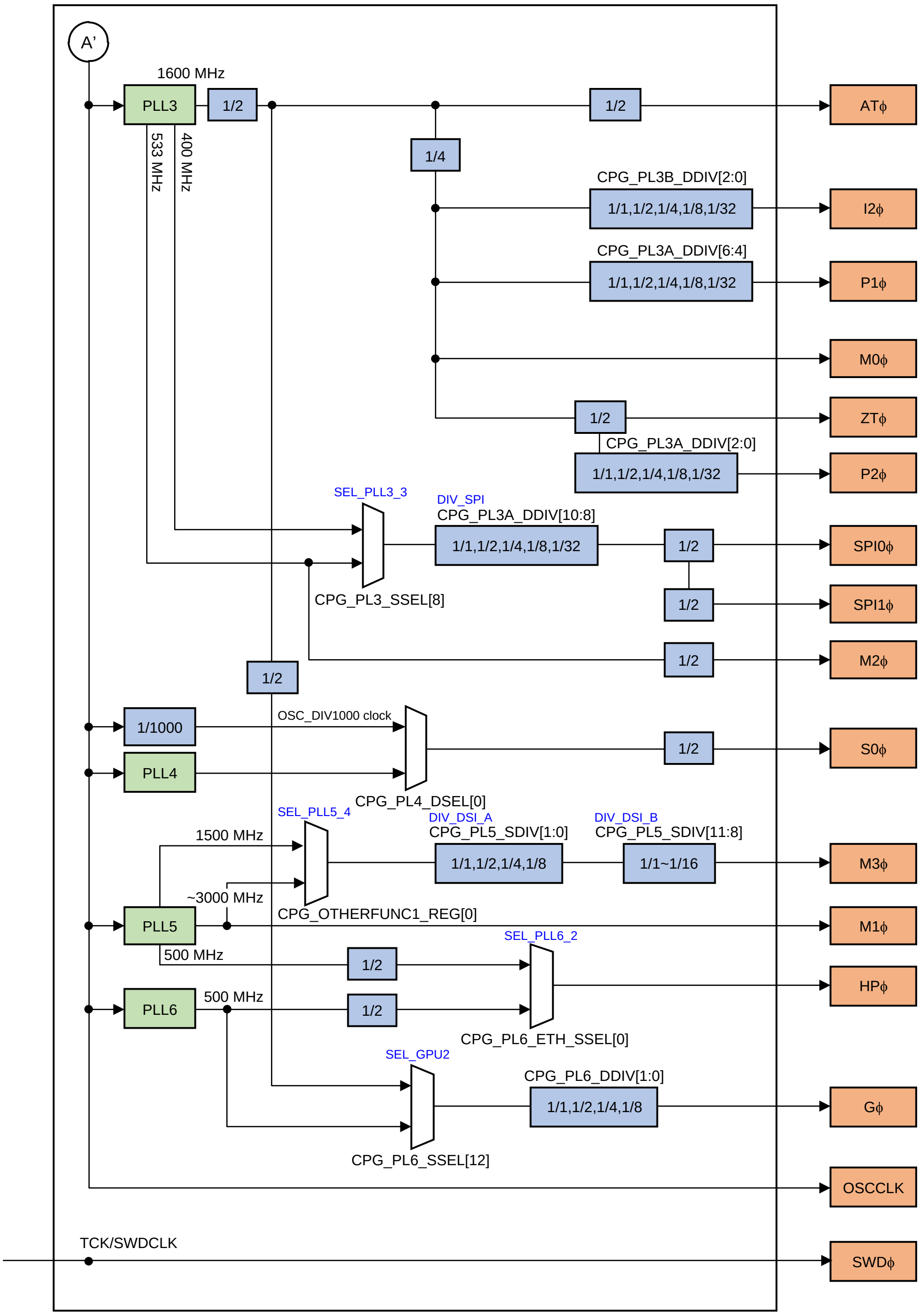
7.1.4.1 Clock System Diagram

Figure 7.2 and **Figure 7.3** show the clock system diagram.



Note: SEL_SDHI0 and SEL_SDHI1 collectively control multiple selectors. The control register for SEL_SDHI0 and SEL_SDHI1 is CPG_PL2SDHI_DSEL.

Figure 7.2 Clock System Diagram (1)



Note: The control registers of each selector are as follows.

- SEL_PLL3_3: CPG_PL3_SSEL
- SEL_PLL4: CPG_PL4_DSEL
- SEL_PLL5_4: CPG_OTHERFUNC1_REG
- SEL_PLL6_2: CPG_PL6_ETH_SSEL
- SEL_GPU2: CPG_PL6_SSEL

Figure 7.3 Clock System Diagram (2)

7.1.4.2 Clock List

Please refer to another file for the clock list.

7.2 Register description

7.2.1 Register attributes

The following table shows the register attributes described in Register Details.

Table 7.3 Register attributes

Register attributes	Description
RW	A register that can be read and written.
RW0	A register that can read and 0b write. 1b Write is invalid.
RW1	A register that can read and 1b write. 0b Write is invalid.
R0W	It is a register that can be written with 0b Read at all times.
R1W	It is a register that can be written with 1b Read at all times.
R	A register that can read. Write is invalid.
R0W0	It is a register that can always read 0b and write 0b. 1b Write is invalid.
R0W1	It is a register that can always read 0b and write 1b. 0b Write is invalid.
R1W0	It is a register that can always read 1b and write 0b. 1b Write is invalid.
R1W1	It is a register that can always read 1b and write 1b. 0b Write is invalid.
RCW0	A register that can be cleared by Read and 0b Write is possible. 1b Write is invalid.
RCW1	A register that can be cleared by Read and 1b Write is possible. 0b Write is invalid.

Note: This LSI does not guarantee the access result to the Reserved bit area and the undefined bit area.
At the same time, the Reserved bit area and undefined area may have values other than the initial value “0”. We do not guarantee the result of changing these values.

7.2.2 Register classification

The table below shows the general classification of the register space in this module.

The address of the CPG register is represented by an offset address from the CPG-based address.

Base Address: H’0_1101_0000 (Cortex-A55 Address Space)
Base Address: H’4101_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’5101_0000 (Cortex-M33 Address Space Secure)

Table 7.4 Register Address space

Classification	Offset Address
SSCG PLL Control Registers (PLL1, 4, 6)	H’000 - H’0FF
Fast PLL Control Registers (PLL2, 3, 5)	H’100 - H’17F
Reserved	H’180 - H’1FF
Frequency Dynamic Change Control Registers	H’200 - H’3FF
Frequency Static Change Control Registers	H’400 - H’4FF
Clock Control Registers	H’500 - H’67F
Clock Monitor Registers	H’680 - H’7FF
Reset Control Registers	H’800 - H’97F
Reset Monitor Registers	H’980 - H’AFF
Other Registers	H’B00 - H’FFF

7.2.3 Register configuration

Table 7.5 Register list (1/5)

Register Name	Abbreviation	R/W	Initial value*1	Address*2	Access Size
PLL1 (SSCG) Standby Control Register	CPG_SAMPLL1_STBY	RW	H’0000 0005	H’000	32
PLL4 (SSCG) Standby Control Register	CPG_SAMPLL4_STBY	RW	H’0000 0004	H’010	32
PLL6 (SSCG) Standby Control Register	CPG_SAMPLL6_STBY	RW	H’0000 0000	H’020	32
PLL1 (SSCG) Output Clock Setting Register 1	CPG_SAMPLL1_CLK1	RW	H’0000 1901	H’004	32
PLL4 (SSCG) Output Clock Setting Register 1	CPG_SAMPLL4_CLK1	RW	H’0000 3203	H’014	32
PLL6 (SSCG) Output Clock Setting Register 1	CPG_SAMPLL6_CLK1	RW	H’0000 3E83	H’024	32
PLL1 (SSCG) Output Clock Setting Register 2	CPG_SAMPLL1_CLK2	RW	H’0018 0601	H’008	32
PLL4 (SSCG) Output Clock Setting Register 2	CPG_SAMPLL4_CLK2	RW	H’0008 2400	H’018	32
PLL6 (SSCG) Output Clock Setting Register 2	CPG_SAMPLL6_CLK2	RW	H’0008 2D02	H’028	32
PLL1 (SSCG) Monitor Register	CPG_SAMPLL1_MON	R	H’0000 0011	H’00C	32
PLL4 (SSCG) Monitor Register	CPG_SAMPLL4_MON	R	H’0000 0000	H’01C	32
PLL6 (SSCG) Monitor Register	CPG_SAMPLL6_MON	R	H’0000 0000	H’02C	32
PLL1_SEL_SETTING Register	CPG_PLL1_SETTING	RW	H’0000 0001	H’040	32
CPG_OTPPLL0_MON Register	CPG_OTPPLL0_MON	R	H’xxxx xxxx	H’044	32
CPG_OTPPLL1_MON Register	CPG_OTPPLL1_MON	R	H’xxxx xxxx	H’048	32
CPG_OTPPLL2_MON Register	CPG_OTPPLL2_MON	R	H’xxxx xxxx	H’04C	32
PLL5 (SSCG) Standby Control Register	CPG_SIPLL5_STBY	RW	H’0000 0014	H’140	32
PLL5 (SSCG) Output Clock Setting Register 1	CPG_SIPLL5_CLK1	RW	H’0000 0111	H’144	32
PLL5 (SSCG) Output Clock Setting Register 2	CPG_SIPLL5_CLK2	RW	H’0000 0100	H’148	32
PLL5 (SSCG) Output Clock Setting Register 3	CPG_SIPLL5_CLK3	RW	H’0000 0006	H’14C	32
PLL5 (SSCG) Output Clock Setting Register 4	CPG_SIPLL5_CLK4	RW	H’007D 00FF	H’150	32
PLL5 (SSCG) Output Clock Setting Register 5	CPG_SIPLL5_CLK5	RW	H’0000 0016	H’154	32
PLL2(SSCG) Monitor Register	CPG_SIPLL2_MON	R	H’0000 0011	H’11C	32
PLL3(SSCG) Monitor Register	CPG_SIPLL3_MON	R	H’0000 0011	H’13C	32
PLL5(SSCG) Monitor Register	CPG_SIPLL5_MON	R	H’0000 0000	H’15C	32
Division Ratio Setting (PLL1) Register	CPG_PL1_DDIV	RW	H’0000 000x	H’200	32
Division Ratio Setting (PLL2) Register	CPG_PL2_DDIV	RW	H’0000 0000	H’204	32
Division Ratio Setting (PLL3A) Register	CPG_PL3A_DDIV	RW	H’0000 0000	H’208	32
Division Ratio Setting (PLL3B) Register	CPG_PL3B_DDIV	RW	H’0000 0000	H’20C	32
Division Ratio Setting (PLL6) Register	CPG_PL6_DDIV	RW	H’0000 0000	H’210	32
Source Clock Setting (SDHI) Register	CPG_PL2SDHI_DSEL	RW	H’0000 0011	H’218	32
Source Clock Setting (DDR) Register	CPG_PL4_DSEL	RW	H’0000 0000	H’21C	32
Clock Status Monitor Register	CPG_CLKSTATUS	R	H’0000 0000	H’280	32
Source Clock Setting Register	CPG_PL3_SSEL	RW	H’0000 0111	H’408	32
Source Clock Setting Register	CPG_PL6_SSEL	RW	H’0000 0101	H’414	32
Source Clock Setting Register	CPG_PL6_ETH_SSEL	RW	H’0000 0010	H’418	32
Division Ratio Setting (PLL5) Register	CPG_PL5_SDIV	RW	H’0000 0000	H’420	32
Clock Control Register Cortex-A55	CPG_CLKON_CA55	RW	H’0000 003F	H’500	32
Clock Control Register Cortex-M33	CPG_CLKON_CM33	RW	H’0000 0000	H’504	32
Clock Control Register ACPU	CPG_CLKON_SRAM_ACPU	RW	H’0000 0001	H’508	32
Clock Control Register MCPU	CPG_CLKON_SRAM_MCPU	RW	H’0000 0001	H’50C	32
Clock Control Register GIC600	CPG_CLKON_GIC600	RW	H’0000 0001	H’514	32

Table 7.5 Register list (2/5)

Register Name	Abbreviation	R/W	Initial value*1	Address*2	Access Size
Clock Control Register IA55	CPG_CLKON_IA55	RW	H’0000 0003	H’518	32
Clock Control Register IM33	CPG_CLKON_IM33	RW	H’0000 0003	H’51C	32
Clock Control Register MHU	CPG_CLKON_MHU	RW	H'0000 0000	H'520	32
Clock Control Register CST	CPG_CLKON_CST	RW	H’0000 07FF	H’524	32
Clock Control Register SYC	CPG_CLKON_SYC	RW	H'0000 0000	H'528	32
Clock Control Register DMAC_REG	CPG_CLKON_DAMC_REG	RW	H’0000 0000	H’52C	32
Clock Control Register GTM	CPG_CLKON_GTM	RW	H’0000 0001	H’534	32
Clock Control Register MTU	CPG_CLKON_MTU	RW	H’0000 0000	H’538	32
Clock Control Register POE3	CPG_CLKON_POE3	RW	H’0000 0000	H’53C	32
Clock Control Register GPT	CPG_CLKON_GPT	RW	H’0000 0000	H’540	32
Clock Control Register POEG	CPG_CLKON_POEG	RW	H’0000 0000	H’544	32
Clock Control Register WDT	CPG_CLKON_WDT	RW	H’0000 00C3	H’548	32
Clock Control Register DDR	CPG_CLKON_DDR	RW	H’0000 0000	H’54C	32
Clock Control Register SPI_MULTI	CPG_CLKON_SPI_MULTI	RW	H’0000 0000	H’550	32
Clock Control Register SDHI	CPG_CLKON_SDHI	RW	H’0000 0000	H’554	32
Clock Control Register 3DGE	CPG_CLKON_3DGE	RW	H’0000 0000	H’558	32
Clock Control Register ISU	CPG_CLKON_ISU	RW	H’0000 0000	H’55C	32
Clock Control Register VCPL4	CPG_CLKON_VCPL4	RW	H’0000 0000	H’560	32
Clock Control Register CRU	CPG_CLKON_CRU	RW	H’0000 0000	H’564	32
Clock Control Register MIPI_DSI	CPG_CLKON_MIPI_DSI	RW	H’0000 0000	H’568	32
Clock Control Register LCDC	CPG_CLKON_LCDC	RW	H’0000 0000	H’56C	32
Clock Control Register SSI	CPG_CLKON_SSI	RW	H’0000 0000	H’570	32
Clock Control Register SRC	CPG_CLKON_SRC	RW	H’0000 0000	H’574	32
Clock Control Register USB	CPG_CLKON_USB	RW	H’0000 0000	H’578	32
Clock Control Register ETH	CPG_CLKON_ETH	RW	H’0000 0000	H’57C	32
Clock Control Register I2C	CPG_CLKON_I2C	RW	H’0000 0000	H’580	32
Clock Control Register SCIF	CPG_CLKON_SCIF	RW	H’0000 0000	H’584	32
Clock Control Register SCI	CPG_CLKON_SCI	RW	H’0000 0000	H’588	32
Clock Control Register IRDA	CPG_CLKON_IRDA	RW	H’0000 0000	H’58C	32
Clock Control Register RSPI	CPG_CLKON_RSPI	RW	H’0000 0000	H’590	32
Clock Control Register CANFD	CPG_CLKON_CANFD	RW	H’0000 0000	H’594	32
Clock Control Register GPIO	CPG_CLKON_GPIO	RW	H’0000 0000	H’598	32
Clock Control Register ADC	CPG_CLKON_ADC	RW	H’0000 0000	H’5A8	32
Clock Control Register TSU	CPG_CLKON_TSU	RW	H’0000 0000	H’5AC	32
Clock Control Register AXI_TZCDDR	CPG_CLKON_AXI_TZCDDR	RW	H’0000 001F	H’5DC	32
Clock Monitor Register Cortex-A55	CPG_CLKMON_CA55	R	H’0000 003F	H’680	32
Clock Monitor Register Cortex-M33	CPG_CLKMON_CM33	R	H’0000 0000	H’684	32
Clock Monitor Register SRAM_ACPU	CPG_CLKMON_SRAM_ACPU	R	H’0000 0001	H’688	32
Clock Monitor Register SRAM_MCPU	CPG_CLKMON_SRAM_MCPU	R	H’0000 0001	H’68C	32
Clock Monitor Register GIC600	CPG_CLKMON_GIC600	R	H’0000 0001	H’694	32
Clock Monitor Register IA55	CPG_CLKMON_IA55	R	H’0000 0003	H’698	32
Clock Monitor Register IM33	CPG_CLKMON_IM33	R	H’0000 0003	H’69C	32
Clock Monitor Register MHU	CPG_CLKMON_MHU	R	H’0000 0000	H’6A0	32
Clock Monitor Register CST	CPG_CLKMON_CST	R	H’0000 07FF	H’6A4	32
Clock Monitor Register SYC	CPG_CLKMON_SYC	R	H’0000 0000	H’6A8	32

Table 7.5 Register list (3/5)

Register Name	Abbreviation	R/W	Initial value*1	Address*2	Access Size
Clock Monitor Register DMAC_REG	CPG_CLKMON_DAMC_REG	R	H’0000 0000	H’6AC	32
Clock Monitor Register GTM	CPG_CLKMON_GTM	R	H’0000 0001	H’6B4	32
Clock Monitor Register MTU	CPG_CLKMON_MTU	R	H’0000 0000	H’6B8	32
Clock Monitor Register POE3	CPG_CLKMON_POE3	R	H’0000 0000	H’6BC	32
Clock Monitor Register GPT	CPG_CLKMON_GPT	R	H’0000 0000	H’6C0	32
Clock Monitor Register POEG	CPG_CLKMON_POEG	R	H’0000 0000	H’6C4	32
Clock Monitor Register WDT	CPG_CLKMON_WDT	R	H’0000 00C3	H’6C8	32
Clock Monitor Register DDR	CPG_CLKMON_DDR	R	H’0000 0000	H’6CC	32
Clock Monitor Register SPI_MULTI	CPG_CLKMON_SPI_MULTI	R	H’0000 0000	H’6D0	32
Clock Monitor Register SDHI	CPG_CLKMON_SDHI	R	H’0000 0000	H’6D4	32
Clock Monitor Register 3DGE	CPG_CLKMON_3DGE	R	H’0000 0000	H’6D8	32
Clock Monitor Register ISU	CPG_CLKMON_ISU	R	H’0000 0000	H’6DC	32
Clock Monitor Register VCPL4	CPG_CLKMON_VCPL4	R	H’0000 0000	H’6E0	32
Clock Monitor Register CRU	CPG_CLKMON_CRU	R	H’0000 0000	H’6E4	32
Clock Monitor Register MIPI_DSI	CPG_CLKMON_MIPI_DSI	R	H’0000 0000	H’6E8	32
Clock Monitor Register LCDC	CPG_CLKMON_LCDC	R	H’0000 0000	H’6EC	32
Clock Monitor Register SSI	CPG_CLKMON_SSI	R	H’0000 0000	H’6F0	32
Clock Monitor Register SRC	CPG_CLKMON_SRC	R	H’0000 0000	H’6F4	32
Clock Monitor Register USB	CPG_CLKMON_USB	R	H’0000 0000	H’6F8	32
Clock Monitor Register ETH	CPG_CLKMON_ETH	R	H’0000 0000	H’6FC	32
Clock Monitor Register I2C	CPG_CLKMON_I2C	R	H’0000 0000	H’700	32
Clock Monitor Register SCIF	CPG_CLKMON_SCIF	R	H’0000 0000	H’704	32
Clock Monitor Register SCI	CPG_CLKMON_SCI	R	H’0000 0000	H’708	32
Clock Monitor Register IRDA	CPG_CLKMON_IRDA	R	H’0000 0000	H’70C	32
Clock Monitor Register RSPI	CPG_CLKMON_RSPI	R	H’0000 0000	H’710	32
Clock Monitor Register CANFD	CPG_CLKMON_CANFD	R	H’0000 0000	H’714	32
Clock Monitor Register GPIO	CPG_CLKMON_GPIO	R	H’0000 0000	H’718	32
Clock Monitor Register ADC	CPG_CLKMON_ADC	R	H’0000 0000	H’728	32
Clock Monitor Register TSU	CPG_CLKMON_TSU	R	H’0000 0000	H’72C	32
Clock Monitor Register AXI_TZCDDR	CPG_CLKMON_AXI_TZCDDR	R	H’0000 001F	H’75C	32
Reset Control Register Cortex-A55	CPG_RST_CA55	RW	H’00001FFF	H’800	32
Reset Control Register Cortex-M33	CPG_RST_CM33	RW	H’0000 0000	H’804	32
Reset Control Register SRAM_ACPU	CPG_RST_SRAM_ACPU	RW	H’0000 0001	H’808	32
Reset Control Register SRAM_MCPU	CPG_RST_SRAM_MCPU	RW	H’0000 0001	H’80C	32
Reset Control Register GIC600	CPG_RST_GIC600	RW	H’0000 0003	H’814	32
Reset Control Register IA55	CPG_RST_IA55	RW	H’0000 0001	H’818	32
Reset Control Register IM33	CPG_RST_IM33	RW	H’0000 0001	H’81C	32
Reset Control Register MHU	CPG_RST_MHU	RW	H’0000 0000	H’820	32
Reset Control Register SYC	CPG_RST_SYC	RW	H’0000 0000	H’828	32
Reset Control Register DMAC	CPG_RST_DMAC	RW	H’0000 0000	H’82C	32
Reset Control Register GTM	CPG_RST_GTM	RW	H’0000 0001	H’834	32
Reset Control Register MTU	CPG_RST_MTU	RW	H’0000 0000	H’838	32
Reset Control Register POE3	CPG_RST_POE3	RW	H’0000 0000	H’83C	32
Reset Control Register GPT	CPG_RST_GPT	RW	H’0000 0000	H’840	32
Reset Control Register POEG	CPG_RST_POEG	RW	H’0000 0000	H’844	32

Table 7.5 Register list (4/5)

Register Name	Abbreviation	R/W	Initial value*[1]	Address*[2]	Access Size
Reset Control Register WDT	CPG_RST_WDT	RW	H’0000 0009	H’848	32
Reset Control Register DDR	CPG_RST_DDR	RW	H’0000 0000	H’84C	32
Reset Control Register SPI	CPG_RST_SPI	RW	H’0000 0000	H’850	32
Reset Control Register SDHI	CPG_RST_SDHI	RW	H’0000 0000	H’854	32
Reset Control Register 3DGE	CPG_RST_3DGE	RW	H’0000 0000	H’858	32
Reset Control Register ISU	CPG_RST_ISU	RW	H’0000 0000	H’85C	32
Reset Control Register VCPL4	CPG_RST_VCPL4	RW	H’0000 0000	H’860	32
Reset Control Register CRU	CPG_RST_CRU	RW	H’0000 0000	H’864	32
Reset Control Register MIPI_DSI	CPG_RST_MIPI_DSI	RW	H’0000 0000	H’868	32
Reset Control Register LCDC	CPG_RST_LCDC	RW	H’0000 0000	H’86C	32
Reset Control Register SSIF	CPG_RST_SSIF	RW	H’0000 0000	H’870	32
Reset Control Register SRC	CPG_RST_SRC	RW	H’0000 0000	H’874	32
Reset Control Register USB	CPG_RST_USB	RW	H’0000 0000	H’878	32
Reset Control Register ETH	CPG_RST_ETH	RW	H’0000 0000	H’87C	32
Reset Control Register I2C	CPG_RST_I2C	RW	H’0000 0000	H’880	32
Reset Control Register SCIF	CPG_RST_SCIF	RW	H’0000 0000	H’884	32
Reset Control Register SCI	CPG_RST_SCI	RW	H’0000 0000	H’888	32
Reset Control Register IRDA	CPG_RST_IRDA	RW	H’0000 0000	H’88C	32
Reset Control Register RSPI	CPG_RST_RSPI	RW	H’0000 0000	H’890	32
Reset Control Register CANFD	CPG_RST_CANFD	RW	H’0000 0000	H’894	32
Reset Control Register GPIO	CPG_RST_GPIO	RW	H’0000 0000	H’898	32
Reset Control Register ADC	CPG_RST_ADC	RW	H’0000 0000	H’8A8	32
Reset Control Register TSU	CPG_RST_TSU	RW	H’0000 0000	H’8AC	32
Reset Control Register AXI_TZCDDR	CPG_RST_AXI_TZCDDR	RW	H’0000 001F	H’8DC	32
Reset Monitor Register Cortex-A55	CPG_RSTMON_CA55	R	H’0000 0000	H’980	32
Reset Monitor Register Cortex-M33	CPG_RSTMON_CM33	R	H’0000 0007	H’984	32
Reset Monitor Register SRAM_ACPU	CPG_RSTMON_SRAM_ACPU	R	H’0000 0000	H’988	32
Reset Monitor Register SRAM_MCPU	CPG_RSTMON_SRAM_MCPU	R	H’0000 0000	H’98C	32
Reset Monitor Register GIC600	CPG_RSTMON_GIC600	R	H’0000 0000	H’994	32
Reset Monitor Register IA55	CPG_RSTMON_IA55	R	H’0000 0000	H’998	32
Reset Monitor Register IM33	CPG_RSTMON_IM33	R	H’0000 0000	H’99C	32
Reset Monitor Register MHU	CPG_RSTMON_MHU	R	H’0000 0001	H’9A0	32
Reset Monitor Register SYC	CPG_RSTMON_SYC	R	H’0000 0001	H’9A8	32
Reset Monitor Register DMAC	CPG_RSTMON_DMAC	R	H’0000 0003	H’9AC	32
Reset Monitor Register GTM	CPG_RSTMON_GTM	R	H’0000 0006	H’9B4	32
Reset Monitor Register MTU	CPG_RSTMON_MTU	R	H’0000 0001	H’9B8	32
Reset Monitor Register POE3	CPG_RSTMON_POE3	R	H’0000 0001	H’9BC	32
Reset Monitor Register GPT	CPG_RSTMON_GPT	R	H’0000 0001	H’9C0	32
Reset Monitor Register POEG	CPG_RSTMON_POEG	R	H’0000 000F	H’9C4	32
Reset Monitor Register WDT	CPG_RSTMON_WDT	R	H’0000 0006	H’9C8	32
Reset Monitor Register DDR	CPG_RSTMON_DDR	R	H’0000 007F	H’9CC	32
Reset Monitor Register SPI	CPG_RSTMON_SPI	R	H’0000 0001	H’9D0	32
Reset Monitor Register SDHI	CPG_RSTMON_SDHI	R	H’0000 0003	H’9D4	32
Reset Monitor Register 3DGE	CPG_RSTMON_3DGE	R	H’0000 0007	H’9D8	32
Reset Monitor Register ISU	CPG_RSTMON_ISU	R	H’0000 0003	H’9DC	32

Table 7.5 Register list (5/5)

Register Name	Abbreviation	R/W	Initial value*1	Address*2	Access Size
Reset Monitor Register VCPL4	CPG_RSTMON_VCPL4	R	H'0000 0003	H'9E0	32
Reset Monitor Register CRU	CPG_RSTMON_CRU	R	H'0000 0007	H'9E4	32
Reset Monitor Register MIPI_DSI	CPG_RSTMON_MIPI_DSI	R	H'0000 0007	H'9E8	32
Reset Monitor Register LCDC	CPG_RSTMON_LCDC	R	H'0000 0001	H'9EC	32
Reset Monitor Register SSIF	CPG_RSTMON_SSIF	R	H'0000 000F	H'9F0	32
Reset Monitor Register SRC	CPG_RSTMON_SRC	R	H'0000 0001	H'9F4	32
Reset Monitor Register USB	CPG_RSTMON_USB	R	H'0000 000F	H'9F8	32
Reset Monitor Register ETH	CPG_RSTMON_ETH	R	H'0000 0003	H'9FC	32
Reset Monitor Register I2C	CPG_RSTMON_I2C	R	H'0000 000F	H'A00	32
Reset Monitor Register SCIF	CPG_RSTMON_SCIF	R	H'0000 001F	H'A04	32
Reset Monitor Register SCI	CPG_RSTMON_SCI	R	H'0000 0003	H'A08	32
Reset Monitor Register IRDA	CPG_RSTMON_IRDA	R	H'0000 0001	H'A0C	32
Reset Monitor Register RSPI	CPG_RSTMON_RSPI	R	H'0000 0007	H'A10	32
Reset Monitor Register CANFD	CPG_RSTMON_CANFD	R	H'0000 0003	H'A14	32
Reset Monitor Register GPIO	CPG_RSTMON_GPIO	R	H'0000 0007	H'A18	32
Reset Monitor Register ADC	CPG_RSTMON_ADC	R	H'0000 0003	H'A28	32
Reset Monitor Register TSU	CPG_RSTMON_TSU	R	H'0000 0001	H'A2C	32
Reset Monitor Register AXI_TZCDDR	CPG_RSTMON_AXI_TZCDDR	R	H'0000 0000	H'A5C	32
WDT Overflow System Reset Register	CPG_WDTOVF_RST	RW	H'0000 0000	H'B10	32
WDT Reset Selector Register	CPG_WDTRST_SEL	RW	H'0000 0088	H'B14	32
Cortex-A55 Cluster Power Status Monitor Register	CPG_CLUSTER_PCHMON	R	H'0000 000x	H'B30	32
Cortex-A55 Cluster Power Status Control Register	CPG_CLUSTER_PCHCTL	RW	H'0048 0000	H'B34	32
Cortex-A55 Core0 Power Status Monitor Register	CPG_CORE0_PCHMON	R	H'0000 000x	H'B38	32
Cortex-A55 Core0 Power Status Control Register	CPG_CORE0_PCHCTL	RW	H'0008 0000	H'B3C	32
Cortex-A55 Core1 Power Status Monitor Register	CPG_CORE1_PCHMON	R	H'0000 000x	H'B40	32
Cortex-A55 Core1 Power Status Control Register	CPG_CORE1_PCHCTL	RW	H'0000 0000	H'B44	32
MSTOP Register ACPU	CPG_BUS_ACPU_MSTOP	RW	H'0000 0000	H'B60	32
MSTOP Register MCPU1	CPG_BUS_MCPU1_MSTOP	RW	H'0000 0000	H'B64	32
MSTOP Register MCPU2	CPG_BUS_MCPU2_MSTOP	RW	H'0000 0000	H'B68	32
MSTOP Register PERI_COM	CPG_BUS_PERI_COM_MSTOP	RW	H'0000 0000	H'B6C	32
MSTOP Register PERI_CPU	CPG_BUS_PERI_CPU_MSTOP	RW	H'0000 0000	H'B70	32
MSTOP Register PERI_DDR	CPG_BUS_PERI_DDR_MSTOP	RW	H'0000 0000	H'B74	32
MSTOP Register PERI_VIDEO	CPG_BUS_PERI_VIDEO_MSTOP	RW	H'0000 0000	H'B78	32
MSTOP Register REG0	CPG_BUS_REG0_MSTOP	RW	H'0000 0000	H'B7C	32
MSTOP Register REG1	CPG_BUS_REG1_MSTOP	RW	H'0000 0000	H'B80	32
MSTOP Register TZCDDR	CPG_BUS_TZCDDR_MSTOP	RW	H'0000 0000	H'B84	32
MSTOP Register MHU	CPG_MHU_MSTOP	RW	H'0000 0000	H'B88	32
Other Function Register 1	CPG_OTHERFUNC1_REG	RW	H'0000 0000	H'BE8	32
Other Function Register 2	CPG_OTHERFUNC2_REG	RW	H'0000 0000	H'BEC	32
Clock Control Register MBUS	CPG_CLKON_MBUS	RW	H'0000 07FF	H'5B8	32
Clock Monitor Register MBUS	CPG_CLKMON_MBUS	R	H'0000 07FF	H'738	32

Note 1. **Initial value:** The reset is released according to the sequence of the system reset, and the initial value is the one immediately after the last reset release in the sequence.

Note 2. **Address:** The offset address from the base address.

7.2.4 Register Descriptions

7.2.4.1 PLLn (SSCG) Standby Control Register (CPG_SAMPLLn_STBY) (n = 1, 4, or 6)

This register is used to control the power-saving mode and standby state and enable or disable the SSCG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	SSCG_MODE_WEN	—	SSCG_EN_WEN	—	RESETB_WEN
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R	R0W1	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SSCG_MODE[1]	SSCG_MODE[0]	—	SSCG_EN	—	RESETB
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	SSCG_MODE_WEN	0b	R0W1	Flag for enabling the writing to the SSCG_MODE[1:0] bits (bits 5 and 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	SSCG_EN_WEN	0b	R0W1	Flag for enabling the writing to the SSCG_EN bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	RESETB_WEN	0b	R0W1	Flag for enabling the writing to the RESETB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	SSCG_MODE[1:0]	00b	RW	SSCG center spread or down spread setting (see the mode setting table below.*1) When writing to these bits, set the SSCG_MODE_WEN bit to 1 at the same time. 00b: Down spread 01b: Setting prohibited 10b: Setting prohibited 11b: Setting prohibited
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	SSCG_EN	n=1,4: 1b n=6: 0b	RW	SSCG enable or disable setting (see the mode setting table below.*1) When writing to this bit, set the SSCG_EN_WEN bit to 1 at the same time. 0: SSCG is disabled. 1: SSCG is enabled.
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RESETB	n=1: 1b n=4,6: 0b	RW	PLL reset setting (see the mode setting table below.*1) When writing to this bit, set the RESETB_WEN bit to 1 at the same time. 0: Reset state 1: Active state (PLL is released from the reset state.)

Note 1. Mode Setting Table

Mode		Bit Name		
		RESETB	SSCG_EN	SSCG_MODE[1:0]
Reset state		0	—	—
SSCG disabled		1	0	—
SSCG enabled	Down spread		1	00b
Setting prohibited				11b

7.2.4.2 PLLn (SSCG) Output Clock Setting Register 1 (CPG_SAMPLLn_CLK1) (n = 1, 4, or 6)

This register is used to specify the frequency division values k, m, and p for SSCG PLLn (n = 1, 4, or 6).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DIV_K															
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DIV_M										DIV_P					
Initial Value n = 1	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	1
Initial Value n = 4	0	0	1	1	0	0	1	0	0	0	0	0	0	0	1	1
Initial Value n = 6	0	0	1	1	1	1	1	0	1	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	DIV_K	All 0	RW	Delta-sigma modulator (DSM) setting*4 Specify a 2’s complement of a decimal value from -32,768 to 32,767 that satisfies the restrictions*1. 1000_0000_0000_0000b: k = -32,768 1000_0000_0000_0001b: k = -32,767 1000_0000_0000_0010b: k = -32,766 : 0111_1111_1111_1101b: k = +32,765 0111_1111_1111_1110b: k = +32,766 0111_1111_1111_1111b: k = +32,767
15 to 6	DIV_M	n = 1: 00_0110_ 0100b n = 4: 00_1100_ 1000b n = 6: 00_1111_ 1010b	RW	Main divider setting*4 Specify an unsigned binary value from 64 to 533 that satisfies the restrictions*1 *2. 11_1111_1111b to 10_0001_0110b (Division value m = 534): Setting prohibited 10_0001_0101b: Division value m = 533 10_0001_0100b: Division value m = 532 : 00_0100_0001b: Division value m = 65 00_0100_0000b: Division value m = 64 00_0011_1111b to 00_0000_0000b: Setting prohibited
5 to 0	DIV_P	‘n = 1: 00_0001b ‘n = 4: 00_0011b ‘n = 6: 00_0011b	RW	Pre-divider setting*4 Specify a value from 1 to 8 that satisfies the restrictions*1 *2. 11_1111b to 00_00_0101b: Setting prohibited 00_0100b: Division value p = 4 (FFREF = 6 MHz) 00_0011:b Division value p = 3 (FFREF = 8 MHz) 00_0010b: Division value p = 2 (FFREF = 12 MHz) 00_0001b: Division value p = 1 (FFREF = 24 MHz) 00_0000b: Setting prohibited

Note 1. The PLL oscillation frequency (F_{FVCO}) and output frequency (F_{FOUT}) can be calculated by the following equations.

- Equations

$$F_{FVCO} = \frac{((m + k/65536) \times F_{Fin})}{p}$$

$$F_{Fout} = \frac{((m + k/65536) \times F_{Fin})}{(p \times 2^{s})}^{*3}$$

- Restrictions

  F_{Fin}: 24 MHz
  6 MHz ≤ F_{FREF} = F_{FIN}/p ≤ 30 MHz
  1,125 MHz ≤ F_{FVCO} ≤ 2,500 MHz

Note 2. Specify DIV_K, DIV_M, DIV_P, and DIV_S*3 within the allowable ranges of F_{FVCO} and F_{Fout} for each PLL shown below.

SSCG PLL	FFout Min.	Default FFout Setting	FFout Max.
PLL1	1,000 MHz	1,200 MHz	1,200 MHz
PLL4	800 MHz	1,600 MHz	1,600 MHz
PLL6	100 MHz	500 MHz	500 MHz

Note 3. For DIV_S (value “s”), see **Section 7.2.4.3, PLLn (SSCG) Output Clock Setting Register 2 (CPG_SAMPLLn_CLK2) (n = 1, 4, or 6)**.

Note 4. The PLL should be reset after the PLL settings are modified. For details, refer to **Section 7.4.5, Procedures for PLL Setup**.

7.2.4.3 PLLn (SSCG) Output Clock Setting Register 2 (CPG_SAMPLLn_CLK2) (n = 1, 4, or 6)

This register is used to specify the frequency division value s and SSCG modulation values mfr and mrr for SSCG PLLn (n = 1, 4, or 6).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	MFR							
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MRR						—	—	—	—	—	DIV_S		
Initial Value n = 1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1
Initial Value n = 4	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	1	0	1	1	0	1	0	0	0	0	0	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	MFR	n = 1: H’18 n = 4: H’08 n = 6: H’08	RW	Modulation frequency setting The PLL should be reset after this setting is modified. For details, refer to **Section 7.4.5, Procedures for PLL Setup**. Specify a value from 0 to 255 (decimal).*1 1111_1111b: mfr = 255 1111_1110b: mfr = 254 : 0000_0001b: mfr = 1 0000_0000b: mfr = 0
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 8	MRR	‘n = 1: 00_0110b ‘n = 4: 10_0100b ‘n = 6: 10_1101b	RW	Modulation ratio setting The PLL should be reset after this setting is modified. For details, refer to **Section 7.4.5, Procedures for PLL Setup**. Specify a value from 1 to 63 (decimal).*1 11_1111b: mrr = 63 11_1110b: mrr = 62 : 00_0010b: mrr = 2 00_0001b: mrr = 1 00_0000b: Setting prohibited
7 to 3	—	00000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	DIV_S	‘n = 1: 001b ‘n = 4: 000b ‘n = 6: 010b	RW	Divider s setting*2 Specify a value from 0 to 6 (decimal). 111b: Setting prohibited 110b: Division value s = 6 (Division ratio = 64) 101b: Division value s = 5 (Division ratio = 32) 100b: Division value s = 4 (Division ratio = 16) 011b: Division value s = 3 (Division ratio = 8) 010b: Division value s = 2 (Division ratio = 4) 001b: Division value s = 1 (Division ratio = 2) 000b: Division value s = 0 (Division ratio = 1)

Note 1. The modulation frequency (MF) and modulation ratio (pk-pk) (MR) can be calculated by the following equations.

- Equations

$$\mathrm{MF} = \frac{(\mathrm{F_{Fin}})}{\mathrm{p} \times \mathrm{mfr} \times (2^5)}$$

$$\mathrm{MR} = \frac{\mathrm{mf} \times \mathrm{mrr}}{\mathrm{m} \times (2^6)} \times 100[\%]$$

- Restrictions

F_{Fin}: 24 MHz
0000_0000b ≤ MFR[7:0] ≤ 1111_1111b
00_0001b ≤ MRR[5:0] ≤ 11_1111b
0 ≤ mrr × mfr ≤ 512

Note 2. For DIV_M (value “m”) and DIV_P (value “p”), see **Section 7.2.4.2, PLLn (SSCG) Output Clock Setting Register 1 (CPG_SAMPLLn_CLK1) (n = 1, 4, or 6)**.

Note 3. Change the SSC setting of PLL4 from the default setting to the following value.

- Modulation Depth [%]: −1.0%
- Fmod [kHz]: About 30 kHz

Important:

For the procedures for setting up the PLL, see **Section 7.4.5, Procedures for PLL Setup**.

Initial Values:

The initial values of the registers for PLL1, PLL4, and PLL6 and output clocks are shown below.

- Initial values of registers

	CPG_SAMPLLn_STBY (n = 1, 4, or 6)		
	SSCG_MODE	SSCG_EN	RESETB
PLL	Bits 5 and 4	Bit 2	Bit 0
PLL1	H’0 (Down spread)	1 (SSCG is enabled)	1 (Released from the reset state)
PLL4	H’0 (Down spread)	1 (SSCG is enabled)	0 (Reset state)
PLL6	H’0 (Down spread)	0 (SSCG is disabled)	0 (Reset state)

	CPG_SAMPLLn_CLK1 (n = 1, 4, or 6)			CPG_SAMPLLn_CLK2 (n = 1, 4, or 6)		
	DIV_K	DIV_M	DIV_P	DIV_S	MFR	MRR
PLL	Bits 31 to 16	Bits 15 to 6	Bits 5 to 0	Bits 2 to 0	Bits 23 to 16	Bits 13 to 8
PLL1	H’0000 (k = 0)	H’064 (m = 100)	H’1 (p = 1)	H’1 (s = 1)	H’18 (mfr = 24)	H’06 (mrr = 6)
PLL4	H’0000 (k = 0)	H’0C8 (m = 200)	H’3 (p = 3)	H’0 (s = 0)	H’08 (mfr = 8)	H’24 (mrr = 36)
PLL6	H’0000 (k = 0)	H’0FA (m = 250)	H’3 (p = 3)	H’2 (s = 2)	H’08 (mfr = 8)	H’2D (mrr = 45)

- Output clocks (initial values)

PLL	PLL Operating Mode	Frequency (F_{Fout}) [MHz]	SSCG	Fmod [kHz]	Spread	Modulation Depth [%]
PLL1	Active state	1,200	Enabled	31.25	Down spread	2.25
PLL4	Reset state	1,600	Enabled	31.25	Down spread	2.25
PLL6	Reset state	500	Disabled	31.25	Down spread	2.25

7.2.4.4 PLLn (SSCG) Monitor Register (CPG_SAMPLLn_MON) (n = 1, 4, or 6)

This register is used to monitor the state of the SSCG PLLn (n = 1, 4, or 6).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PLLn_LOCK	—	—	—	PLLn_RESETB
Initial Value n = 1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
Initial Value n = 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PLLn_LOCK	n = 1: 1b n = 4: 0b n = 6: 0b	R	SSCG PLL lock state monitoring 0: PLL is not locked. 1: PLL is locked.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PLLn_RESETB	n = 1: 1b n = 4: 0b n = 6: 0b	R	SSCG PLL operating mode monitoring 1: Normal mode 0: Reset state (standby mode)

7.2.4.5 PLL1_SEL_SETTING Register (CPG_PLL1_SETTING)

This register is used to specify the frequency of PLL1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL1_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	SEL_PLL1_WEN	0b	R0W1	Flag for enabling the writing to the SEL_PLL1 bit. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SEL_PLL1	1b	RW	The PLL multiplication factor is specified. 0: PLL1 operates according to the settings in the PLL1 SYSREG of the CPG. 1: When bit 0 of OTP_OTPPLL0 is 1, PLL1 operates according to the settings of OTP_OTPPLL0 to OTP_OTPPLL2. When bit 0 of OTP_OTPPLL0 is 0, PLL1 operates according to the settings in the PLL1 SYSREG of the CPG.

7.2.4.6 CPG_OTPPLL0_MON Register (CPG_OTPPLL0_MON)

This register indicates the state of the OTP_OTPPLL0[31:0] pins of the CPG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTP31_ MON	OTP30_ MON	OTP29_ MON	OTP28_ MON	OTP27_ MON	OTP26_ MON	OTP25_ MON	OTP24_ MON	OTP23_ MON	OTP22_ MON	OTP21_ MON	OTP20_ MON	OTP19_ MON	OTP18_ MON	OTP17_ MON	OTP16_ MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTP15_ MON	OTP14_ MON	OTP13_ MON	OTP12_ MON	OTP11_ MON	OTP10_ MON	OTP9_ MON	OTP8_ MON	OTP7_ MON	OTP6_ MON	OTP5_ MON	OTP4_ MON	OTP3_ MON	OTP2_ MON	OTP1_ MON	OTP0_ MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OTPn_MON [n = 31 to 0]	—	R	These bits indicate the state of the OTP_OTPPLL0[31:0] pins of the CPG. *Note*: The values specified in the OTP are set in these bits as the initial values.

7.2.4.7 CPG_OTPPLL1_MON Register (CPG_OTPPLL1_MON)

This register indicates the state of the OTP_OTPPLL1[31:0] pins of the CPG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTP31_ MON	OTP30_ MON	OTP29_ MON	OTP28_ MON	OTP27_ MON	OTP26_ MON	OTP25_ MON	OTP24_ MON	OTP23_ MON	OTP22_ MON	OTP21_ MON	OTP20_ MON	OTP19_ MON	OTP18_ MON	OTP17_ MON	OTP16_ MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTP15_ MON	OTP14_ MON	OTP13_ MON	OTP12_ MON	OTP11_ MON	OTP10_ MON	OTP9_ MON	OTP8_ MON	OTP7_ MON	OTP6_ MON	OTP5_ MON	OTP4_ MON	OTP3_ MON	OTP2_ MON	OTP1_ MON	OTP0_ MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OTPn_MON [n = 31 to 0]	—	R	These bits indicate the state of the OTP_OTPPLL1[31:0] pins of the CPG. *Note*: The values specified in the OTP are set in these bits as the initial values.

7.2.4.8 CPG_OTPPLL2_MON Register (CPG_OTPPLL2_MON)

This register indicates the state of the OTP_OTPPLL2[31:0] pins of the CPG.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTP31_MON	OTP30_MON	OTP29_MON	OTP28_MON	OTP27_MON	OTP26_MON	OTP25_MON	OTP24_MON	OTP23_MON	OTP22_MON	OTP21_MON	OTP20_MON	OTP19_MON	OTP18_MON	OTP17_MON	OTP16_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTP15_MON	OTP14_MON	OTP13_MON	OTP12_MON	OTP11_MON	OTP10_MON	OTP9_MON	OTP8_MON	OTP7_MON	OTP6_MON	OTP5_MON	OTP4_MON	OTP3_MON	OTP2_MON	OTP1_MON	OTP0_MON
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OTPn_MON [n = 31 to 0]	—	R	These bits indicate the state of the OTP_OTPPLL2[31:0] pins of the CPG. *Note*: The values specified in the OTP are set in these bits as the initial values.

7.2.4.9 PLL5 (SSCG) Standby Control Register (CPG_SIPLL5_STBY)

This register is used to control the power-saving mode and standby state and enable or disable the SSC in PLL5 that operates at 3 GHz or a higher frequency. When enable PLL5, change to SSCG_EN=0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	DOWNSPREAD_WEN	—	SSCG_EN_WEN	—	RESETB_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R	R0W1	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DOWNSPREAD	—	SSCG_EN	—	RESETB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	DOWNSPREAD_WEN	0b	R0W1	Flag for enabling the writing to the DOWNSPREAD bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	SSCG_EN_WEN	0b	R0W1	Flag for enabling the writing to the SSCG_EN bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	RESETB_WEN	0b	R0W1	Flag for enabling the writing to the RESETB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	DOWNSPREAD	1b	RW	SSC center spread or down spread setting When writing to this bit, set the DOWNSPREAD_WEN bit to 1 at the same time. 0: Setting prohibited 1: Down spread
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	SSCG_EN*[2]	1b	RW	SSC enable or disable setting (see the mode setting table below.*[1]) When writing to this bit, set the SSCG_EN_WEN bit to 1 at the same time. 0: SSC is disabled. 1: SSC is enabled.
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	RESETB	0b	RW	PLL reset setting (see the mode setting table below.*[1]) When writing to this bit, set the RESETB_WEN bit to 1 at the same time. 0: Reset state 1: Active state (PLL is released from the reset state.)

Note 1. Mode Setting Table

Note 2. Set the SSCG_EN bit to 0, when operating PLL5.

Mode		Bit Name		
		RESETB	SSCG_EN	DOWNSPREAD
Reset state		0	—	—
SSCG disabled		1	0	—
SSCG enabled	Down spread		1	1b

7.2.4.10 PLL5 (SSCG) Output Clock Setting Register 1 (CPG_SIPLL5_CLK1)

This register is used to set up PLL5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	REFDIV						—	POSTDIV2			—	POSTDIV1		
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 8	REFDIV	00_0001b	RW	REFDIV setting for PLL Specify a value from 1 to 63.
7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	POSTDIV2	001b	RW	POSTDIV2 setting for PLL Specify a value from 1 to 7. The POSTDIV value is obtained by POSTDIV1 × POSTDIV2.
3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	POSTDIV1	001b	RW	POSTDIV1 setting for PLL Specify a value from 1 to 7. The POSTDIV value is obtained by POSTDIV1 × POSTDIV2.

7.2.4.11 PLL5 (SSCG) Output Clock Setting Register 2 (CPG_SIPLL5_CLK2)

This register is used to set up PLL5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FOUTVCOPD_WEN	—	FOUTPOSTDIVPD_WEN	—	FOUT4PHASEPD_WEN	—	DSMPD_WEN	—	DACPD_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R	R0W1	R	R0W1	R	R0W1	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	FOUTVCOPD	—	FOUTPOSTDIVPD	—	FOUT4PHASEPD	—	DSMPD	—	DACPD
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	RW	R	RW	R	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	FOUTVCOPD_WEN	0b	R0W1	Flag for enabling the writing to the FOUTVCOPD bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	FOUTPOSTDIVPD_WEN	0b	R0W1	Flag for enabling the writing to the FOUTPOSTDIVPD bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	FOUT4PHASEPD_WEN	0b	R0W1	Flag for enabling the writing to the FOUT4PHASEPD bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	DSMPD_WEN	0b	R0W1	Flag for enabling the writing to the DSMPD bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DACPD_WEN	0b	R0W1	Flag for enabling the writing to the DACPD bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
8	FOUTVCOPD	1b	RW	FOUTVCOPD setting for the PLL Power-down of FOUTVCO is specified. 0: FOUTVCO is active. 1: FOUTVCO power is turned down.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	FOUTPOSTDIVPD	0b	RW	FOUTPOSTDIVPD setting for the PLL Power-down of FOUTPOSTDIVPD is specified. (Target: FOUTPOSTDIVPD, FOUT1PH0, FOUT2, FOUT3, and FOUT4) 0: FOUTPOSTDIVPD is active. 1: FOUTPOSTDIVPD power is turned down.
5	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	FOUT4PHASEPD	0b	RW	FOUT4PHASEPD setting for the PLL Power-down of FOUT4PHASEPD is specified. (Target: FOUT1PH0, FOUT2, FOUT3, and FOUT4) 0: FOUT4PHASEPD is active. 1: FOUT4PHASEPD power is turned down.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	DSMPD	0b	RW	Delta-sigma modulator (DSM) power-down setting 0: DSM is active. 1: DSM power is turned down.
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	DACPD	0b	RW	Noise-cancelling DAC power-down setting in the FRAC mode 0: DAC is active (default mode). 1: DAC power is turned down (for test mode only).

7.2.4.12 PLL5 (SSCG) Output Clock Setting Register 3 (CPG_SIPLL5_CLK3)

This register is used to set up the SSCG for PLL5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FRACIN															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FRACIN								—	—	DIVVAL					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	FRACIN	H’00_0000	RW	Division value setting for frequency division
7, 6	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	DIVVAL	00_0110b	RW	Modulation frequency setting When DIVVAL = 6’d0, the division value is set to 1. When DIVVAL > 0, the DIVVAL value is used as the division value.

7.2.4.13 PLL5 (SSCG) Output Clock Setting Register 4 (CPG_SIPLL5_CLK4)

This register is used to set up the SSCG for PLL5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	INTIN											
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
27 to 16	INTIN	H’07D	RW	Setting of integer part of division value.
15 to 8	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 0	—	H’FF	RW	Reserved When read, the initial value is read. The written value will be ignored.

7.2.4.14 PLL5 (SSCG) Output Clock Setting Register 5 (CPG_SIPLL5_CLK5)

This register is used to set up the SSCG for PLL5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SPREAD				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	SPREAD	1_0110b	RW	Spread depth setting 00000b: Setting prohibited 00001b: 0.1% 00010b: 0.2% : 10000b: 1.6% : 11111b: 3.1%

Important:

For the procedures for setting up the PLL, see **Section 7.4.5, Procedures for PLL Setup**.

Initial Values:

The initial values of the registers for PLL5 and output clock are shown below.

- Initial values of registers

	CPG_SIPLL5_STBY		
	DOWNSPREAD	SSCG_EN	RESETB
PLL	Bit 4	Bit 2	Bit 0
PLL5	H'1 (Down spread)	1 (SSCG is enabled)	0 (Reset state)

	CPG_SIPLL5_CLK1		
	REFDIV	POSTDIV2	POSTDIV1
PLL	Bits 13 to 8	Bits 6 to 4	Bits 2 to 0
PLL5	H'01 (REFDIV = 1)	H'1 (POSTDIV2 = 1)	H'1 (POSTDIV1 = 1)

	CPG_SIPLL5_CLK3	
	FRACIN	DIVVAL
PLL	Bits 31 to 8	Bits 5 to 0
PLL5	H'00_0000 (FRACIN = 0)	H'6 (DIVVAL = 6)

	CPG_SIPLL5_CLK4	CPG_SIPLL5_CLK5
	INTIN	SPREAD
PLL	Bits 27 to 16	Bits 4 to 0
PLL5	H'07D (INTIN = 125)	H'16 (SPREAD = 22)

- Output clock (initial values)

PLL	PLL Operating Mode	Frequency (F_{Fout}) [MHz]	SSCG	Fmod [kHz]	Spread	Modulation Depth [%]
PLL5	Reset state	3,000	Enabled	31.25	Down spread	2.20

FOUTVCO = FREF/REFDIV × (INTIN + FRACIN / 2^24)

FOUTPOSTDIV = FOUTVCO / (POSTDIV1 × POSTDIV2)

The specifiable ranges of parameters for PLL5 are shown below.

Parameter	Restrictions	Specifiable Range
FREF	Fixed to the OSC frequency (24)	24
REFDIV	10 ≤ FREF/REFDIV ≤ FOUTVCO/20 (Frac)	1 to 2
INTIN*[1]	20 to 320 (Frac) and 800 ≤ FOUTVCO ≤ 3000 (PLL5)	Values that satisfy the restriction shown to the left
FRACIN*[1]	800 ≤ FOUTVCO ≤ 3000 (PLL5)	Values that satisfy the restriction shown to the left
POSTDIV1	—	1 to 7
POSTDIV2	—	1 to 7

Note 1. The restrictions when the SSCG is disabled are shown. When the SSCG is enabled, the restrictions should be satisfied with the modulation depth taken into account.

7.2.4.15 PLLn (SSCG) Monitor Register (CPG_SIPLLn_MON) (n = 2, 3, or 5)

This register is used to monitor the state of the SSCG PLLn (n = 2, 3, or 5).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value n = 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial Value n = 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PLLn_LOCK	—	—	—	PLLn_RESETB
Initial Value n = 2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
Initial Value n = 3	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
Initial Value n = 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PLLn_LOCK [n = 2, 3, or 5]	n = 2: 1b n = 3: 1b n = 5: 0b	R	SSCG PLL lock state monitoring 0: PLL is not locked. 1: PLL is locked.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PLLn_RESETB [n = 2, 3, or 5]	n = 2: 1b n = 3: 1b n = 5: 0b	R	SSCG PLL operating mode monitoring 1: Normal mode 0: Reset state (standby mode)

7.2.4.16 Division Ratio Setting (PLL1) Register (CPG_PL1_DDIV)

This register is used to set the division ratio of the clock for the Cortex-A55.

The PLL1 (1,200 MHz) output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIV_PLL1SET_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIVPL1_SET	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. Writing to these bits is ignored.
16	DIV_PLL1SET_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL1_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved These bits are always read as 0. Writing to these bits is ignored.
1, 0	DIVPL1_SET	00b	RW	Iϕ division ratio setting 00b: 1/1 (1,200MHz) 01b: 1/2 (600MHz) 10b: 1/4 (300MHz) 11b: 1/8 (150MHz)

7.2.4.17 Division Ratio Setting (PLL2) Register (CPG_PL2_DDIV)

This register is used to set the division ratios of the clocks for the system bus and the IP modules that cannot use SSCG clocks.

The PLL2 (up to 3,200 MHz) output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DIV_DSI_LPCLK_WEN	—	—	—	—	—	—	—	—	—	—	—	DIV_PLL2_A_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R0W1	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DIVDSILPCLK_SET		—	—	—	—	—	—	—	—	—	DIVPL2A_SET		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	DIV_DSI_LPCLK_WEN	0b	R0W1	Flag for enabling the writing to the DIVDSILPCLK_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27 to 17	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIV_PLL2_A_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL2A_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13, 12	DIVDSILPCLK_SET	00b	RW	M4ϕ division ratio setting 00b: 1/16 (16.656MHz) 01b: 1/32 (8.328MHz) 10b: 1/64 (4.164MHz) 11b: 1/128 (2.082MHz)
11	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10 to 8	—	000b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	—	000b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	DIVPL2A_SET	000b	RW	P0ϕ division ratio setting 000b: 1/1 (100MHz) 001b: 1/2 (50MHz) 010b: 1/4 (25MHz) 011b: 1/8 (12.5MHz) 100b: 1/32 (3.125MHz) Others: Setting prohibited

7.2.4.18 Division Ratio Setting (PLL3A) Register (CPG_PL3A_DDIV)

This register is used to set the division ratios of the clocks for the system bus and the IP modules that can use SSCG clocks.

The clock obtained by dividing the output of PLL3 is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	DIV_PLL3_C_WEN	—	—	—	DIV_PLL3_B_WEN	—	—	—	DIV_PLL3_A_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R	R	R	R0W1	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DIVPL3C_SET			—	DIVPL3B_SET			—	DIVPL3A_SET		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	DIV_PLL3_C_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL3C_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 21	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	DIV_PLL3_B_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL3B_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19 to 17	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIV_PLL3_A_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL3A_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10 to 8	DIVPL3C_SET	000b	RW	DIV_PLL3_C division ratio setting 000b: 1/1 (400/533MHz)*[1] 001b: 1/2 (200/266.5MHz)*[1] 010b: 1/4 (100/133.25MHz)*[1] 011b: 1/8 (50/66.625MHz)*[1] 100b: 1/32 (12.5/16.656MHz)*[1] Others: Setting prohibited *Note 1.* (xx/yyMHz) The source clock for xx is 400MHz. The source clock for yy is 533MHz. Please select 400 MHz or 533MHz with CPG_PL3_SSEL Register. In the latter stage, SPI0ϕ is divided into 1/2 and SPI1ϕ is divided into 1/4

Bit	Bit Name	Initial Value	R/W	Description
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	DIVPL3B_SET	000b	RW	P1ϕ division ratio setting The source clock uses a clock (200MHz) that is 1/8 of the output of PLL3 (1,600MHz). 000b: 1/1 (200MHz) 001b: 1/2 (100MHz) 010b: 1/4 (50MHz) 011b: 1/8 (25MHz) 100b: 1/32 (6.25MHz) Others: Setting prohibited
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	DIVPL3A_SET	000b	RW	P2ϕ division ratio setting The source clock uses a clock (100MHz) that is 1/16 of the output of PLL3 (1,600MHz). 000b: 1/1 (100MHz) 001b: 1/2 (50MHz) 010b: 1/4 (25MHz) 011b: 1/8 (12.5MHz) 100b: 1/32 (3.125MHz) Others: Setting prohibited

7.2.4.19 Division Ratio Setting (PLL3B) Register (CPG_PL3B_DDIV)

This register is used to set the division ratio of the clock for the Cortex-M33.

The source clock uses a clock (200MHz) that is 1/8 of the output of PLL3 (1,600MHz).

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIV_PLL3_CLK200FIX_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DIVPL3CLK200FIX_SET		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIV_PLL3_CLK200FIX_WEN	0b	R0W1	Flag for enabling the writing to the DIVPL3CLK200FIX_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	DIVPL3CLK200FIX_SET	000b	RW	I2ϕ division ratio setting 000b: 1/1 (200MHz) 001b: 1/2 (100MHz) 010b: 1/4 (50MHz) 011b: 1/8 (25MHz) 100b: 1/32 (6.25MHz) Others: Setting prohibited *Note*: Before writing to this field, read the value at first.It is possible to write to this field only if the reading value is different from the writing value.

7.2.4.20 Division Ratio Setting (PLL6) Register (CPG_PL6_DDIV)

This register is used to set the division ratios of the clocks for the 3DGE.

The PLL6 (up to 500 MHz) output is used as the source clock.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIV_GPU_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DIVGPU_SET	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIV_GPU_WEN	0b	R0W1	Flag for enabling the writing to the DIVGPU_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	DIVGPU_SET	00b	RW	Gϕ division ratio setting 00b: 1/1(400/500MHz)*1 01b: 1/2(200/250MHz)*1 10b: 1/4(100/125MHz)*1 11b: 1/8(50/62.5MHz)*1 *Note 1.* (xx/yyMHz) The source clock for xx is 400MHz. The source clock for yy is 500MHz. Please select 400MHz or 500MHz with CPG_PL6_SSEL Register.

7.2.4.21 Source Clock Setting (SDHI) Register (CPG_PL2SDHI_DSEL)

This register is used to switch the clocks for the SDHI.

The setting of this register can be dynamically modified while the clock is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	SEL_SDHI1_WEN	—	—	—	SEL_SDHI0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SEL_SDHI1_SET		—	—	SEL_SDHI0_SET	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	SEL_SDHI1_WEN	0b	R0W1	Flag for enabling the writing to the SEL_SDHI1_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	SEL_SDHI0_WEN	0b	R0W1	Flag for enabling the writing to the SEL_SDHI0_SET bits This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5, 4	SEL_SDHI1_SET	01b	RW	SEL_SDHI1 clock switching*[1] 00b: Setting prohibited 01b: 533 MHz 10b: 400 MHz 11b: 266 MHz
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	SEL_SDHI0_SET	01b	RW	SEL_SDHI0 clock switching*[1] 00b: Setting prohibited 01b: 533 MHz 10b: 400 MHz 11b: 266 MHz

Note: When 1 is written to the SEL_SDHI1_WEN or SEL_SDHI0_WEN bit placed in the upper 16 bits of this register, the clock switching control begins even if the settings in the lower bits are not changed. Check the status monitor register and wait until the switching is completed. The clock will temporarily stop at the timing of switching.

Note 1. To change the setting from 01b (533 MHz) to 10b (400 MHz) or vice versa, do not directly switch to the target clock. Switch to 11b (266 MHz) first, and then switch to the target setting (01b (533 MHz) or 10b (400 MHz)). Switching the clock directly between the above frequencies can temporarily generate pulses faster than 533MHz, which can cause malfunctions.

7.2.4.22 Source Clock Setting (DDR) Register (CPG_PL4_DSEL)

This register is used to switch the source clock for the DDR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL4_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL4_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	SEL_PLL4_WEN	0b	R0W1	Flag for enabling the writing to the SEL_PLL4_SET bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SEL_PLL4_SET	0b	RW	Source clock setting 0: OSC_DIV1000 clock (0.024MHz) 1: PLL4 clock (1600MHz) *Note*: The clock selected by this register is halved in the latter stage. 0.024MHz / 2 = 0.012MHz 1600MHz / 2 = 800MHz S0ϕ = 0.012MHz or 800MHz

7.2.4.23 Clock Status Monitor Register (CPG_CLKSTATUS)

This register is used to monitor whether clock switching is completed in the dynamically modifiable frequency dividers and selectors.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SELPLL4_STS	—	SELSDHI1_STS	SELSDHI0_STS	—	—	—	—	—	—	—	DIVGPU_STS	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DIVPL3CLK200FIX_STS	DIVPL3C_STS	DIVPL3B_STS	DIVPL3A_STS	DIVDSILPCLK_STS	—	—	DIVPL2A_STS	—	—	—	DIVPL1_STS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SELPLL4_STS	0b	R	SELPLL4 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
30	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
29	SELSDHI1_STS	0b	R	SELSDHI1 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
28	SELSDHI0_STS	0b	R	SELSDHI0 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
27 to 24	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
23 to 21	—	000b	R	Reserved When read, the initial value is read. The written value will be ignored.
20	DIVGPU_STS	0b	R	DIVGPU clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
19 to 12	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
11	DIVPL3CLK200FIX_STS	0b	R	DIVPL3CLK200FIX clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
10	DIVPL3C_STS	0b	R	DIVPL3C clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
9	DIVPL3B_STS	0b	R	DIVPL3B clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
8	DIVPL3A_STS	0b	R	DIVPL3A clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).

Bit	Bit Name	Initial Value	R/W	Description
7	DIVDSILPCLK_STS	0b	R	DIVDSILPCLK clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
6	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
5	—	0b	R	Reserved When read, the initial value is read. The written value will be ignored.
4	DIVPL2A_STS	0b	R	DIVPL2A clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).
3 to 1	—	000b	R	Reserved When read, the initial value is read. The written value will be ignored.
0	DIVPL1_STS	0b	R	DIVPL1 clock switch completion state 0: Switching is completed. 1: Switching is not completed (busy).

7.2.4.24 Source Clock Setting Register (CPG_PL3_SSEL)

This register is used to switch the source clocks for the system bus and the IPs that can use SSCG clocks.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SEL_PLL3_3_WEN	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SEL_PLL3_3_SET	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	SEL_PLL3_3_WEN	0b	RW	Flag for enabling the writing to the SEL_PLL3_3_SET bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	SEL_PLL3_3_SET	1b	RW	Source clock setting 0: 533MHz 1: 400MHz
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

7.2.4.25 Source Clock Setting Register (CPG_PL6_SSEL)

This register is used to switch the source clocks for the 3DGE.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SEL_GPU2_WEN	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R0W1	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SEL_GPU2_SET	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	SEL_GPU2_WEN	0b	R0W1	Flag for enabling the writing to the SEL_GPU2_SET bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	SEL_GPU2_SET	0b	RW	Source clock setting 0: 500MHz 1: 400MHz
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

7.2.4.26 Source Clock Setting Register (CPG_PL6_ETH_SSEL)

This register is used to switch the source clocks for the Gigabit Ethernet Interface.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL6_2_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEL_PLL6_2_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	SEL_PLL6_2_WEN	0b	R0W1	Flag for enabling the writing to the SEL_PLL6_2_SET bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 1	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SEL_PLL6_2_SET	0b	RW	Source clock setting 0: PLL6 clock (250MHz) 1: PLL5 clock (250MHz)

7.2.4.27 Division Ratio Setting (PLL5) Register (CPG_PL5_SDIV)

This register is used to set the division ratio of the clock for the MIPI DSI.

The clock obtained by dividing the PLL5 (Up to 3000 MHz or 1500MHz) output is used as the source clock.

Select Up to 3000 MHz or 1500 MHz with SEL_PLL5_4 in the previous stage.

This register does not support dynamic modification.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	DIV_DSI_B_WEN	—	—	—	—	—	—	—	DIV_DSI_A_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DIVDSIB_SET				—	—	—	—	—	—	DIVDSIA_SET	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	DIV_DSI_B_WEN	0b	R0W1	Flag for enabling the writing to the DIVDSIB_SET bits (bits 11 to 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DIV_DSI_A_WEN	0b	R0W1	Flag for enabling the writing to the DIVDSIA_SET bits (bits 1 and 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	DIVDSIB_SET	0000b	RW	Division ratio setting for the MIPI clock The division ratio is obtained by DIV_DSI_A × DIV_DSI_B and these bits specify DIV_DSI_B. The clock output range is 5.08 to 148.5MHz. 0000b: 1/1 0001b: 1/2 : 1110b: 1/15 1111b: 1/16
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	DIVDSIA_SET	00b	RW	Division ratio setting for the MIPI clock The division ratio is obtained by DIV_DSI_A × DIV_DSI_B and these bits specify DIV_DSI_A. 00b: 1/1 01b: 1/2 10b: 1/4 11b: 1/8

Note: The DIV_DSI_A and DIV_DSI_B settings should satisfy the following conditions.

Mode	DIV_DSI_A				DIV_DSI_B
	Input Clock (SEL_PLL5_4 Route Setting) Selection by CPG_OTHERFUNC1_REG (Bit 0) 0: FOUTPOSTDIV 1: FOUT1PH0	Maximum Input Frequency	Division Setting	Maximum Output Frequency	Division Setting
DSI (MIPI_DSI_PLLCLK/MIPI_DSI_VCLK*1 = even value)	1: FOUT1PH0	1.5 GHz	1/1, 1/2, 1/4, or 1/8	750 MHz	1/1 to 1/16
DSI (MIPI_DSI_PLLCLK/MIPI_DSI_VCLK*1 = odd value)	0: FOUTPOSTDIV	750 MHz	1	750 MHz	Odd value
DPI	0: FOUTPOSTDIV	167 MHz	1/2, 1/4, or 1/8	83.5 MHz	1/1

Note 1. For MIPI_DSI_VCLK, refer to the separate clock list.

7.2.4.28 Clock Control Register Cortex-A55 (CPG_CLKON_CA55)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_ON	1b	RW	The CA55_TSCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	1b	RW	The CA55_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
3	CLK3_ON	1b	RW	The CA55_GICCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	The CA55_ATCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The CA55_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The CA55_SCLK/PERICLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.29 Clock Control Register Cortex-M33 (CPG_CLKON_CM33)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The CM33_TSCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The CM33_CLKIN clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.30 Clock Control Register ACPU (CPG_CLKON_SRAM_ACPU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	1b	RW	The SRAM_ACPU_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.31 Clock Control Register MCPU (CPG_CLKON_SRAM_MCPU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	1b	RW	The SRAM_MCPU_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.32 Clock Control Register GIC600 (CPG_CLKON_GIC600)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	1b	RW	The GIC600_GICCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.33 Clock Control Register IA55 (CPG_CLKON_IA55)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	The IA55_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The IA55_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.34 Clock Control Register IM33 (CPG_CLKON_IM33)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	1b	RW	The IM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The IM33_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.35 Clock Control Register MHU (CPG_CLKON_MHU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The MHU_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.36 Clock Control Register CST (CPG_CLKON_CST)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CLK10_ONWEN	CLK9_ONWEN	CLK8_ONWEN	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CLK10_ON	CLK9_ON	CLK8_ON	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	CLK10_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	CLK8_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	CLK10_ON	1b	RW	The CST_AXI_ETR_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
9	CLK9_ON	1b	RW	The CST_AXI_SB_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
8	CLK8_ON	1b	RW	The CST_ATB_CM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
7	CLK7_ON	1b	RW	The CST_ATB_CA55_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	1b	RW	The CST_AHB_ATH_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	1b	RW	The CST_AHB_CM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	1b	RW	The CST_APB_CA55_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	1b	RW	The CST_APB_CM33_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	The CST_APB_SB_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The CST_TS_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The CST_CS_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.37 Clock Control Register SYC (CPG_CLKON_SYC)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The SYC_CNT_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.38 Clock Control Register DMAC_REG (CPG_CLKON_DMAC_REG)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The DMAC_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The DMAC_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.39 Clock Control Register GTM (CPG_CLKON_GTM)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CLK2_ON	0b	RW	The OSTM2_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The OSTM1_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The OSTM0_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.40 Clock Control Register MTU (CPG_CLKON_MTU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The MTU_X_MCK_MTU3 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.41 Clock Control Register POE3 (CPG_CLKON_POE3)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The POE3_CLKM_POE clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.42 Clock Control Register GPT (CPG_CLKON_GPT)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The GPT_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.43 Clock Control Register POEG (CPG_CLKON_POEG)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The POEG_D_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The POEG_C_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The POEG_B_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The POEG_A_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.44 Clock Control Register WDT (CPG_CLKON_WDT)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
5	CLK5_ON	0b	RW	The WDT2_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
4	CLK4_ON	0b	RW	The WDT2_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The WDT1_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The WDT1_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The WDT0_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	1b	RW	The WDT0_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.45 Clock Control Register DDR (CPG_CLKON_DDR)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The DDR_REG_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The DDR_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.46 Clock Control Register SPI_MULTI (CPG_CLKON_SPI_MULTI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The SPI_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SPI_CLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.47 Clock Control Register SDHI (CPG_CLKON_SDHI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	CLK7_ON	0b	RW	The SDHI1_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	0b	RW	The SDHI1_CLK_HS clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	0b	RW	The SDHI1_IMCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	0b	RW	The SDHI1_IMCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The SDHI0_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The SDHI0_CLK_HS clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The SDHI0_IMCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SDHI0_IMCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.48 Clock Control Register 3DGE (CPG_CLKON_3DGE)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CLK2_ON	0b	RW	The GPU_ACE_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The GPU_AXI_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The GPU_CLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.49 Clock Control Register ISU (CPG_CLKON_ISU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The ISU_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The ISU_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.50 Clock Control Register VCPL4 (CPG_CLKON_VCPL4)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The H264_CLK_A and H264_CLK_P clocks are supplied or stopped. 0: Clocks are stopped. 1: Clocks are supplied.

7.2.4.51 Clock Control Register CRU (CPG_CLKON_CRU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	CLK3_ON	0b	RW	The CRU_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The CRU_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The CRU_VCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
0	CLK0_ON	0b	RW	The CRU_SYSCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.52 Clock Control Register MIPI_DSI (CPG_CLKON_MIPI_DSI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_ON	0b	RW	The MIPI_DSI_LPCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	0b	RW	The MIPI_DSI_VCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
3	CLK3_ON	0b	RW	The MIPI_DSI_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The MIPI_DSI_ACLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The MIPI_DSI_SYSCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The MIPI_DSI_PLLCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.53 Clock Control Register LCDC (CPG_CLKON_LCDC)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The LCDC_CLK_D clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The LCDC_CLK_A and LCDC_CLK_P clocks are supplied or stopped. 0: Clocks are stopped. 1: Clocks are supplied.

7.2.4.54 Clock Control Register SSI (CPG_CLKON_SSI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	CLK7_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	CLK6_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	CLK5_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	CLK7_ON	0b	RW	The SSI3_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
6	CLK6_ON	0b	RW	The SSI3_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
5	CLK5_ON	0b	RW	The SSI2_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
4	CLK4_ON	0b	RW	The SSI2_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The SSI1_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The SSI1_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The SSI0_PCLK_SFR clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SSI0_PCLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.55 Clock Control Register SRC (CPG_CLKON_SRC)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The SRC_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.56 Clock Control Register USB (CPG_CLKON_USB)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The USB_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The USB_U2P_EXR_CPUCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The USB_U2H1_HCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The USB_U2H0_HCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.57 Clock Control Register ETH (CPG_CLKON_ETH)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The ETH1_CLK_AXI/CHI clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The ETH0_CLK_AXI/CHI clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.58 Clock Control Register I2C (CPG_CLKON_I2C)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_ON	0b	RW	The I2C3_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The I2C2_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The I2C1_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The I2C0_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.59 Clock Control Register SCIF (CPG_CLKON_SCIF)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CLK4_ON	0b	RW	The SCIF4_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	0b	RW	The SCIF3_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	0b	RW	The SCIF2_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The SCIF1_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
0	CLK0_ON	0b	RW	The SCIF0_CLK_PCK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.60 Clock Control Register SCI (CPG_CLKON_SCI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The SCI1_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The SCI0_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.61 Clock Control Register IRDA (CPG_CLKON_IRDA)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The IRDA_CLKP clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.62 Clock Control Register RSPI (CPG_CLKON_RSPI)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CLK2_ON	0b	RW	The RSPI2_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	0b	RW	The RSPI1_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The RSPI0_CLKB clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.63 Clock Control Register CANFD (CPG_CLKON_CANFD)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The CANFD_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.64 Clock Control Register GPIO (CPG_CLKON_GPIO)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The GPIO_HCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.65 Clock Control Register ADC (CPG_CLKON_ADC)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_ON	0b	RW	The ADC_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
0	CLK0_ON	0b	RW	The ADC_ADCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.66 Clock Control Register TSU (CPG_CLKON_TSU)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_ON	0b	RW	The TSU_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.67 Clock Control Register AXI_TZCDDR (CPG_CLKON_AXI_TZCDDR)

This register is used to supply or stop clocks for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	CLK4_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	CLK3_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	CLK2_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	CLK1_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	CLK0_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CLK4_ON	1b	RW	The BUS_TZCDDR_ACLK3 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
3	CLK3_ON	1b	RW	The BUS_TZCDDR_ACLK2 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2	CLK2_ON	1b	RW	The BUS_TZCDDR_ACLK1 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
1	CLK1_ON	1b	RW	The BUS_TZCDDR_ACLK0 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

Bit	Bit Name	Initial Value	R/W	Description
0	CLK0_ON	1b	RW	The BUS_TZCDDR_PCLK clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.

7.2.4.68 Clock Monitor Register Cortex-A55 (CPG_CLKMON_CA55)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ MON	CLK4_ MON	CLK3_ MON	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_MON	1b	R	The state of the CA55_TSCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	1b	R	The state of the CA55_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	The state of the CA55_GICCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	The state of the CA55_ATCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the CA55_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the CA55_SCLK/PERICLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.69 Clock Monitor Register Cortex-M33 (CPG_CLKMON_CM33)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the CM33_TSCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the CM33_CLKIN clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.70 Clock Monitor Register SRAM_ACPU (CPG_CLKMON_SRAM_ACPU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	1b	R	The state of the SRAM_ACPU_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.71 Clock Monitor Register SRAM_MCPU (CPG_CLKMON_SRAM_MCPU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	1b	R	The state of the SRAM_MCPU_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.72 Clock Monitor Register GIC600 (CPG_CLKMON_GIC600)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	1b	R	The state of the GIC600_GICCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.73 Clock Monitor Register IA55 (CPG_CLKMON_IA55)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	The state of the IA55_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the IA55_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.74 Clock Monitor Register IM33 (CPG_CLKMON_IM33)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	1b	R	The state of the IM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the IM33_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.75 Clock Monitor Register MHU (CPG_CLKMON_MHU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the MHU_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.76 Clock Monitor Register CST (CPG_CLKMON_CST)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CLK10_MON	CLK9_MON	CLK8_MON	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	CLK10_MON	1b	R	The state of the CST_AXI_ETR_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
9	CLK9_MON	1b	R	The state of the CST_AXI_SB_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
8	CLK8_MON	1b	R	The state of the CST_ATB_CM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
7	CLK7_MON	1b	R	The state of the CST_ATB_CA55_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	1b	R	The state of the CST_AHB_ATH_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	1b	R	The state of the CST_AHB_CM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	1b	R	The state of the CST_APB_CA55_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	The state of the CST_APB_CM33_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	The state of the CST_APB_SB_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the CST_TS_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the CST_CS_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

Note: No CGC is provided for CST_SWCLKTCK.

7.2.4.77 Clock Monitor Register SYC (CPG_CLKMON_SYC)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the SYC_CNT_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.78 Clock Monitor Register DMAC_REG (CPG_CLKMON_DMAC_REG)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the DMAC_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the DMAC_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.79 Clock Monitor Register GTM (CPG_CLKMON_GTM)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CLK2_MON	0b	R	The state of the OSTM2_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the OSTM1_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the OSTM0_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.80 Clock Monitor Register MTU (CPG_CLKMON_MTU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the MTU_X_MCK_MTU3 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.81 Clock Monitor Register POE3 (CPG_CLKMON_POE3)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the POE3_CLKM_POE clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.82 Clock Monitor Register GPT (CPG_CLKMON_GPT)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the GPT_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.83 Clock Monitor Register POEG (CPG_CLKMON_POEG)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the POEG_D_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the POEG_C_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the POEG_B_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the POEG_A_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.84 Clock Monitor Register WDT (CPG_CLKMON_WDT)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_ MON	CLK4_ MON	CLK3_ MON	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	—	1b	R	Reserved When read, the initial value is read. The written value will be ignored.
6	—	1b	R	Reserved When read, the initial value is read. The written value will be ignored.
5	CLK5_MON	0b	R	The state of the WDT2_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the WDT2_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the WDT1_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the WDT1_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the WDT0_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the WDT0_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.85 Clock Monitor Register DDR (CPG_CLKMON_DDR)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the DDR_REG_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the DDR_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.86 Clock Monitor Register SPI_MULTI (CPG_CLKMON_SPI_MULTI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the SPI_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SPI_CLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.87 Clock Monitor Register SDHI (CPG_CLKMON_SDHI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_ MON	CLK6_ MON	CLK5_ MON	CLK4_ MON	CLK3_ MON	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CLK7_MON	0b	R	The state of the SDHI1_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	0b	R	The state of the SDHI1_CLK_HS clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	0b	R	The state of the SDHI1_IMCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the SDHI1_IMCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the SDHI0_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the SDHI0_CLK_HS clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the SDHI0_IMCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SDHI0_IMCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.88 Clock Monitor Register 3DGE (CPG_CLKMON_3DGE)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored..
2	CLK2_MON	0b	R	The state of the GPU_ACE_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the GPU_AXI_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the GPU_CLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.89 Clock Monitor Register ISU (CPG_CLKMON_ISU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the ISU_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the ISU_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.90 Clock Monitor Register VCPL4 (CPG_CLKMON_VCPL4)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the H264_CLK_A/P clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.91 Clock Monitor Register CRU (CPG_CLKMON_CRU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ MON	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	CLK3_MON	0b	R	The state of the CRU_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the CRU_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the CRU_VCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the CRU_SYSCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.92 Clock Monitor Register MIPI_DSI (CPG_CLKMON_MIPI_DSI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CLK5_MON	0b	R	The state of the MIPI_DSI_LPCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the MIPI_DSI_VCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the MIPI_DSI_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the MIPI_DSI_ACLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the MIPI_DSI_SYSCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the MIPI_DSI_PLLCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.93 Clock Monitor Register LCDC (CPG_CLKMON_LCDC)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the LCDC_CLK_D clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the LCDC_CLK_A/P clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.94 Clock Monitor Register SSI (CPG_CLKMON_SSI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK7_ MON	CLK6_ MON	CLK5_ MON	CLK4_ MON	CLK3_ MON	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CLK7_MON	0b	R	The state of the SSI3_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
6	CLK6_MON	0b	R	The state of the SSI3_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
5	CLK5_MON	0b	R	The state of the SSI2_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
4	CLK4_MON	0b	R	The state of the SSI2_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the SSI1_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the SSI1_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the SSI0_PCLK_SFR clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SSI0_PCLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.95 Clock Monitor Register SRC (CPG_CLKMON_SRC)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the SRC_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.96 Clock Monitor Register USB (CPG_CLKMON_USB)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the USB_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the USB_U2P_EXR_CPUCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the USB_U2H1_HCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the USB_U2H0_HCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.97 Clock Monitor Register ETH (CPG_CLKMON_ETH)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the ETH1_CLK_AXI/CHI clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the ETH0_CLK_AXI/CHI clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

Note: No CGC is provided for ETH1_REFCLK and ETH0_REFCLK.

7.2.4.98 Clock Monitor Register I2C (CPG_CLKMON_I2C)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLK3_ MON	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	CLK3_MON	0b	R	The state of the I2C3_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the I2C2_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the I2C1_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the I2C0_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.99 Clock Monitor Register SCIF (CPG_CLKMON_SCIF)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CLK4_MON	0b	R	The state of the SCIF4_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	0b	R	The state of the SCIF3_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	0b	R	The state of the SCIF2_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the SCIF1_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SCIF0_CLK_PCK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.100 Clock Monitor Register SCI (CPG_CLKMON_SCI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored..
1	CLK1_MON	0b	R	The state of the SCI1_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the SCI0_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.101 Clock Monitor Register IRDA (CPG_CLKMON_IRDA)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the IRDA_CLKP clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.102 Clock Monitor Register RSPI (CPG_CLKMON_RSPI)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK2_ MON	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	CLK2_MON	0b	R	The state of the RSPI2_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	0b	R	The state of the RSPI1_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the RSPI0_CLKB clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.103 Clock Monitor Register CANFD (CPG_CLKMON_CANFD)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the CANFD_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.104 Clock Monitor Register GPIO (CPG_CLKMON_GPIO)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the GPIO_HCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.105 Clock Monitor Register ADC (CPG_CLKMON_ADC)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK1_ MON	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CLK1_MON	0b	R	The state of the ADC_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	0b	R	The state of the ADC_ADCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.106 Clock Monitor Register TSU (CPG_CLKMON_TSU)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLK0_MON	0b	R	The state of the TSU_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.107 Clock Monitor Register AXI_TZCDDR (CPG_CLKMON_AXI_TZCDDR)

This register is used to monitor the state of the clocks supplied to individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CLK4_MON	1b	R	The state of the BUS_TZCDDR_ACLK3 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
3	CLK3_MON	1b	R	The state of the BUS_TZCDDR_ACLK2 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
2	CLK2_MON	1b	R	The state of the BUS_TZCDDR_ACLK1 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
1	CLK1_MON	1b	R	The state of the BUS_TZCDDR_ACLK0 clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.
0	CLK0_MON	1b	R	The state of the BUS_TZCDDR_PCLK clock is monitored. 0: Clock is not supplied. 1: Clock is supplied.

7.2.4.108 Reset Control Register Cortex-A55 (CPG_RST_CA55)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	UNIT12_RSTWEN	UNIT11_RSTWEN	UNIT10_RSTWEN	UNIT9_RSTWEN	UNIT8_RSTWEN	UNIT7_RSTWEN	UNIT6_RSTWEN	UNIT5_RSTWEN	UNIT4_RSTWEN	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	UNIT12_RSTB	UNIT11_RSTB	UNIT10_RSTB	UNIT9_RSTB	UNIT8_RSTB	UNIT7_RSTB	UNIT6_RSTB	UNIT5_RSTB	UNIT4_RSTB	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	UNIT12_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT12_RSTB bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	UNIT11_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT11_RSTB bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	UNIT10_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT10_RSTB bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	UNIT9_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT9_RSTB bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	UNIT8_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT8_RSTB bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	UNIT7_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT7_RSTB bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	UNIT6_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT6_RSTB bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	UNIT5_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT5_RSTB bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
20	UNIT4_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	UNIT12_RSTB	1b	RW	The CA55_RST12 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
11	UNIT11_RSTB	1b	RW	The CA55_RST11 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
10	UNIT10_RSTB	1b	RW	The CA55_RST10 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
9	UNIT9_RSTB	1b	RW	The CA55_RST9 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
8	UNIT8_RSTB	1b	RW	The CA55_RST8 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
7	UNIT7_RSTB	1b	RW	The CA55_RST7 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
6	UNIT6_RSTB	1b	RW	The CA55_RST6reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
5	UNIT5_RSTB	1b	RW	The CA55_RST5 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
4	UNIT4_RSTB	1b	RW	The CA55_RST4 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	UNIT3_RSTB	1b	RW	The CA55_RST3_1 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
2	UNIT2_RSTB	1b	RW	The CA55_RST3_0 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	1b	RW	The CA55_RST1_1 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The CA55_RST1_0 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.109 Reset Control Register Cortex-M33 (CPG_RST_CM33)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_ RST WEN	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_ RSTB	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The CM33_MISCRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The CM33_NSYSRESET reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The CM33_NPORESET reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.110 Reset Control Register SRAM_ACPU (CPG_RST_SRAM_ACPU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	The SRAM_ACPU_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.111 Reset Control Register SRAM_MCPU (CPG_RST_SRAM_MCPU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	The SRAM_MCPU_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.112 Reset Control Register GIC600 (CPG_RST_GIC600)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	1b	RW	The GIC600_DBG_GICRESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The GIC600_GICRESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.113 Reset Control Register IA55 (CPG_RST_IA55)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	The IA55_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.114 Reset Control Register IM33 (CPG_RST_IM33)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	1b	RW	The IM33_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.115 Reset Control Register MHU (CPG_RST_MHU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The MHU_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.116 Reset Control Register SYC (CPG_RST_SYC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The SYC_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.117 Reset Control Register DMAC (CPG_RST_DMAC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The DMAC_RST_ASYNC reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The DMAC_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.118 Reset Control Register GTM (CPG_RST_GTM)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_ RST WEN	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_ RSTB	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The OSTM2_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The OSTM1_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The OSTM0_PRESETZ reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.119 Reset Control Register MTU (CPG_RST_MTU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The MTU_X_PRESET_MTU3 reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.120 Reset Control Register POE3 (CPG_RST_POE3)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The POE3_RST_M_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.121 Reset Control Register GPT (CPG_RST_GPT)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The GPT_RST_C reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.122 Reset Control Register POEG (CPG_RST_POEG)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RST WEN	UNIT2_RST WEN	UNIT1_RST WEN	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The POEG_D_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The POEG_C_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The POEG_B_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The POEG_A_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.123 Reset Control Register WDT (CPG_RST_WDT)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	—	1b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	UNIT2_RSTB	0b	RW	The WDT2_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The WDT1_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	1b	RW	The WDT0_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.124 Reset Control Register DDR (CPG_RST_DDR)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	UNIT6_ RST WEN	UNIT5_ RST WEN	UNIT4_ RST WEN	UNIT3_ RST WEN	UNIT2_ RST WEN	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	UNIT6_ RSTB	UNIT5_ RSTB	UNIT4_ RSTB	UNIT3_ RSTB	UNIT2_ RSTB	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	UNIT6_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT6_RSTB bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	UNIT5_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT5_RSTB bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	UNIT4_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	UNIT6_RSTB	0b	RW	The DDR_REG_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
5	UNIT5_RSTB	0b	RW	The DDR_AXI3_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
4	UNIT4_RSTB	0b	RW	The DDR_AXI2_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	UNIT3_RSTB	0b	RW	The DDR_AXI1_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The DDR_AXI0_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The DDR_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The DDR_RESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.125 Reset Control Register SPI (CPG_RST_SPI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The SPI_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.126 Reset Control Register SDHI (CPG_RST_SDHI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RST WEN	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The SDHI1_IXRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SDHI0_IXRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.127 Reset Control Register 3DGE (CPG_RST_3DGE)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The GPU_ACE_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The GPU_AXI_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The GPU_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.128 Reset Control Register ISU (CPG_RST_ISU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The ISU_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The ISU_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.129 Reset Control Register VCPL4 (CPG_RST_VCPL4)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RST WEN	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The H264_CP_PRESET_P reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The H264_X_RESET_VCP reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.130 Reset Control Register CRU (CPG_RST_CRU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The CRU_ARESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The CRU_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The CRU_CMN_RSTB reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.131 Reset Control Register MIPI_DSI (CPG_RST_MIPI_DSI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The MIPI_DSI_PRESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The MIPI_DSI_ARESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The MIPI_DSI_CMN_RSTB reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.132 Reset Control Register LCDC (CPG_RST_LCDC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The LCDC_RESET_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.133 Reset Control Register SSIF (CPG_RST_SSIF)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The SSI3_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The SSI2_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The SSI1_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SSI0_RST_M2_REG reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.134 Reset Control Register SRC (CPG_RST_SRC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The SRC_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.135 Reset Control Register USB (CPG_RST_USB)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The USB_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The USB_U2P_EXL_SYSRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The USB_U2H1_HRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The USB_U2H0_HRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.136 Reset Control Register ETH (CPG_RST_ETH)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The ETH1_RST_HW_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The ETH0_RST_HW_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.137 Reset Control Register I2C (CPG_RST_I2C)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_ RST WEN	UNIT2_ RST WEN	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UNIT3_ RSTB	UNIT2_ RSTB	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	UNIT3_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	UNIT3_RSTB	0b	RW	The I2C3_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The I2C2_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The I2C1_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The I2C0_MRST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.138 Reset Control Register SCIF (CPG_RST_SCIF)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	UNIT4_ RST WEN	UNIT3_ RST WEN	UNIT2_ RST WEN	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	UNIT4_ RSTB	UNIT3_ RSTB	UNIT2_ RSTB	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	UNIT4_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	UNIT4_RSTB	0b	RW	The SCIF4_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	UNIT3_RSTB	0b	RW	The SCIF3_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	0b	RW	The SCIF2_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The SCIF1_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
0	UNIT0_RSTB	0b	RW	The SCIF0_RST_SYSTEM_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.139 Reset Control Register SCI (CPG_RST_SCI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RST WEN	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The SCI1_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The SCI0_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.140 Reset Control Register IRDA (CPG_RST_IRDA)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The IRDA_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.141 Reset Control Register RSPI (CPG_RST_RSPI)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_ RST WEN	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_ RSTB	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The RSPI2_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The RSPI1_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The RSPI0_RST reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.142 Reset Control Register CANFD (CPG_RST_CANFD)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_ RST WEN	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_ RSTB	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The CANFD_RSTC_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The CANFD_RSTP_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.143 Reset Control Register GPIO (CPG_RST_GPIO)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RST WEN	UNIT1_RST WEN	UNIT0_RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UNIT2_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	UNIT2_RSTB	0b	RW	The GPIO_SPARE_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	0b	RW	The GPIO_PORT_RESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The GPIO_RSTN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.144 Reset Control Register ADC (CPG_RST_ADC)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	UNIT1_RSTB	0b	RW	The ADC_ADRST_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
0	UNIT0_RSTB	0b	RW	The ADC_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.145 Reset Control Register TSU (CPG_RST_TSU)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RST WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNIT0_ RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNIT0_RST WEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UNIT0_RSTB	0b	RW	The TSU_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.146 Reset Control Register AXI_TZCDDR (CPG_RST_AXI_TZCDDR)

This register is used to control the reset signals for individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	UNIT4_RSTWEN	UNIT3_RSTWEN	UNIT2_RSTWEN	UNIT1_RSTWEN	UNIT0_RSTWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	UNIT4_RSTB	UNIT3_RSTB	UNIT2_RSTB	UNIT1_RSTB	UNIT0_RSTB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	UNIT4_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT4_RSTB bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	UNIT3_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT3_RSTB bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	UNIT2_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT2_RSTB bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	UNIT1_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT1_RSTB bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	UNIT0_RSTWEN	0b	R0W1	Flag for enabling the writing to the UNIT0_RSTB bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	UNIT4_RSTB	1b	RW	The BUS_TZCDDR_ARESET3N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
3	UNIT3_RSTB	1b	RW	The BUS_TZCDDR_ARESET2N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
2	UNIT2_RSTB	1b	RW	The BUS_TZCDDR_ARESET1N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).
1	UNIT1_RSTB	1b	RW	The BUS_TZCDDR_ARESET0N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Bit	Bit Name	Initial Value	R/W	Description
0	UNIT0_RSTB	1b	RW	The BUS_TZCDDR_PRESETN reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

7.2.4.147 Reset Monitor Register Cortex-A55 (CPG_RSTMON_CA55)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RST12_MON	RST11_MON	RST10_MON	RST9_MON	RST8_MON	RST7_MON	RST6_MON	RST5_MON	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	RST12_MON	0b	R	The state of the CA55_RST12 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
11	RST11_MON	0b	R	The state of the CA55_RST11 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
10	RST10_MON	0b	R	The state of the CA55_RST10 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
9	RST9_MON	0b	R	The state of the CA55_RST9 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
8	RST8_MON	0b	R	The state of the CA55_RST8 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
7	RST7_MON	0b	R	The state of the CA55_RST7 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
6	RST6_MON	0b	R	The state of the CA55_RST6 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
5	RST5_MON	0b	R	The state of the CA55_RST5 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
4	RST4_MON	0b	R	The state of the CA55_RST4 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	0b	R	The state of the CA55_RST3_1 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	0b	R	The state of the CA55_RST3_0 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

Bit	Bit Name	Initial Value	R/W	Description
1	RST1_MON	0b	R	The state of the CA55_RST1_1 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the CA55_RST1_0 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.148 Reset Monitor Register Cortex-M33 (CPG_RSTMON_CM33)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the CM33_MISCRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the CM33_NSYSRESET reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the CM33_NPORESET reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.149 Reset Monitor Register SRAM_ACPU (CPG_RSTMON_SRAM_ACPU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the SRAM_ACPU_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.150 Reset Monitor Register SRAM_MCPU (CPG_RSTMON_SRAM_MCPU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the SRAM_MCPU_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.151 Reset Monitor Register GIC600 (CPG_RSTMON_GIC600)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	0b	R	The state of the GIC600_DBG_GICRESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the GIC600_GICRESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.152 Reset Monitor Register IA55 (CPG_RSTMON_IA55)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the IA55_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.153 Reset Monitor Register IM33 (CPG_RSTMON_IM33)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	0b	R	The state of the IM33_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.154 Reset Monitor Register MHU (CPG_RSTMON_MHU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the MHU_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.155 Reset Monitor Register SYC (CPG_RSTMON_SYC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the SYC_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.156 Reset Monitor Register DMAC (CPG_RSTMON_DMAC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the DMAC_RST_ASYNC reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the DMAC_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.157 Reset Monitor Register GTM (CPG_RSTMON_GTM)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the OSTM2_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the OSTM1_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the OSTM0_PRESETZ reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.158 Reset Monitor Register MTU (CPG_RSTMON_MTU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the MTU_X_PRESET_MTU3 reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.159 Reset Monitor Register POE3 (CPG_RSTMON_POE3)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the POE3_RST_M_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.160 Reset Monitor Register GPT (CPG_RSTMON_GPT)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the GPT_RST_C reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.161 Reset Monitor Register POEG (CPG_RSTMON_POEG)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_ MON	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the POEG_D_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the POEG_C_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the POEG_B_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the POEG_A_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.162 Reset Monitor Register WDT (CPG_RSTMON_WDT)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the WDT2_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the WDT1_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the WDT0_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.163 Reset Monitor Register DDR (CPG_RSTMON_DDR)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RST6_MON	RST5_MON	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	RST6_MON	1b	R	The state of the DDR_REG_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
5	RST5_MON	1b	R	The state of the DDR_AXI3_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
4	RST4_MON	1b	R	The state of the DDR_AXI2_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	1b	R	The state of the DDR_AXI1_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the DDR_AXI0_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the DDR_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the DDR_RESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.164 Reset Monitor Register SPI (CPG_RSTMON_SPI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the SPI_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.165 Reset Monitor Register SDHI (CPG_RSTMON_SDHI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the SDHI1_IXRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SDHI0_IXRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.166 Reset Monitor Register 3DGE (CPG_RSTMON_3DGE)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the GPU_ACE_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the GPU_AXI_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the GPU_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.167 Reset Monitor Register ISU (CPG_RSTMON_ISU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the ISU_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the ISU_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.168 Reset Monitor Register VCPL4 (CPG_RSTMON_VCPL4)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the H264_CP_PRESET_P reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the H264_X_RESET_VCP reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.169 Reset Monitor Register CRU (CPG_RSTMON_CRU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the CRU_ARESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the CRU_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the CRU_CMN_RSTB reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.170 Reset Monitor Register MIPI_DSI (CPG_RSTMON_MIPI_DSI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the MIPI_DSI_PRESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the MIPI_DSI_ARESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the MIPI_DSI_CMN_RSTB reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.171 Reset Monitor Register LCDC (CPG_RSTMON_LCDC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the LCDC_RESET_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.172 Reset Monitor Register SSIF (CPG_RSTMON_SSIF)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_ MON	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the SSI3_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the SSI2_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the SSI1_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SSI0_RST_M2_REG reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.173 Reset Monitor Register SRC (CPG_RSTMON_SRC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the SRC_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.174 Reset Monitor Register USB (CPG_RSTMON_USB)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_ MON	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the USB_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the USB_U2P_EXL_SYSRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the USB_U2H1_HRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the USB_U2H0_HRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.175 Reset Monitor Register ETH (CPG_RSTMON_ETH)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the ETH1_RST_HW_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the ETH0_RST_HW_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.176 Reset Monitor Register I2C (CPG_RSTMON_I2C)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RST3_ MON	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	RST3_MON	1b	R	The state of the I2C3_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the I2C2_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the I2C1_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the I2C0_MRST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.177 Reset Monitor Register SCIF (CPG_RSTMON_SCIF)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	RST4_MON	1b	R	The state of the SCIF4_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	1b	R	The state of the SCIF3_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	1b	R	The state of the SCIF2_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the SCIF1_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SCIF0_RST_SYSTEM_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.178 Reset Monitor Register SCI (CPG_RSTMON_SCI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the SCI1_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the SCI0_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.179 Reset Monitor Register IRDA (CPG_RSTMON_IRDA)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the IRDA_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.180 Reset Monitor Register RSPI (CPG_RSTMON_RSPI)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_ MON	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the RSPI2_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the RSPI1_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the RSPI0_RST reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.181 Reset Monitor Register CANFD (CPG_RSTMON_CANFD)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the CANFD_RSTC_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the CANFD_RSTP_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.182 Reset Monitor Register GPIO (CPG_RSTMON_GPIO)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RST2_MON	1b	R	The state of the GPIO_SPARE_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	1b	R	The state of the GPIO_PORT_RESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the GPIO_RSTN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.183 Reset Monitor Register ADC (CPG_RSTMON_ADC)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST1_ MON	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RST1_MON	1b	R	The state of the ADC_ADRST_N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	1b	R	The state of the ADC_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.184 Reset Monitor Register TSU (CPG_RSTMON_TSU)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST0_ MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RST0_MON	1b	R	The state of the TSU_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.185 Reset Monitor Register AXI_TZCDDR (CPG_RSTMON_AXI_TZCDDR)

This register is used to monitor the reset signals of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	RST4_MON	0b	R	The state of the BUS_TZCDDR_ARESET3N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
3	RST3_MON	0b	R	The state of the BUS_TZCDDR_ARESET2N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
2	RST2_MON	0b	R	The state of the BUS_TZCDDR_ARESET1N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
1	RST1_MON	0b	R	The state of the BUS_TZCDDR_ARESET0N reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).
0	RST0_MON	0b	R	The state of the BUS_TZCDDR_PRESETN reset signal is monitored. 0: Reset signal is not applied (released from the reset state). 1: Reset signal is applied (reset state).

7.2.4.186 WDT Overflow System Reset Register (CPG_WDTOVF_RST)

After the release from a system reset applied by the WDT reset circuit in response to a reset request such as a WDT overflow, this register can be used to identify the WDT channel that generated the reset request. If a system reset is not applied according to the WDTRSTSEL register setting, the WDT is not reset and the source of the reset request can be identified by checking the interrupt from the WDT and the status register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTOVF2_WEN	WDTOVF1_WEN	WDTOVF0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTOVF2	WDTOVF1	WDTOVF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	WDTOVF2_WEN	0b	R0W1	Flag for enabling the writing to the WDTOVF2 bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	WDTOVF1_WEN	0b	R0W1	Flag for enabling the writing to the WDTOVF1 bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	WDTOVF0_WEN	0b	R0W1	Flag for enabling the writing to the WDTOVF0 bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	WDTOVF2	0b	RW	The system reset generated by WDT channel 2 for the Cortex-M33 is indicated. 0: System reset has not been generated by WDT channel 2 for the Cortex-M33. 1: System reset has been generated by WDT channel 2 for the Cortex-M33. *Note*: This bit is cleared by writing 1.
1	WDTOVF1	0b	RW	The system reset generated by WDT channel 1 for Cortex-A55 Core 1 is indicated. 0: System reset has not been generated by WDT channel 1 for Cortex-A55 Core 1. 1: System reset has been generated by WDT channel 1 for Cortex-A55 Core 1. *Note*: This bit is cleared by writing 1.
0	WDTOVF0	0b	RW	The system reset generated WDT channel 0 for Cortex-A55 Core 0 is indicated. 0: System reset has not been generated by WDT channel 0 for Cortex-A55 Core 0. 1: System reset has been generated by WDT channel 0 for Cortex-A55 Core 0. *Note*: This bit is cleared by writing 1.

7.2.4.187 WDT Reset Selector Register (CPG_WDTRST_SEL)

This register is used to mask reset requests from the WDT.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	WDTRSTSEL10	WDTRSTSEL9	WDTRSTSEL8	—	WDTRSTSEL6	WDTRSTSEL5	WDTRSTSEL4	—	WDTRSTSEL2	WDTRSTSEL1	WDTRSTSEL0
Initial Value	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	WDTRSTSEL 10	0b	RW	Whether to mask the reset request by WDT channel 2 is specified. 0: Cold reset of the Cortex-M33 is masked. 1: Cold reset of the Cortex-M33 is enabled.
9	WDTRSTSEL 9	0b	RW	Whether to mask the reset request by WDT channel 1 is specified. 0: Cold reset of the Cortex-A55 is masked. 1: Cold reset of the Cortex-A55 is enabled.
8	WDTRSTSEL 8	0b	RW	Whether to mask the reset request by WDT channel 0 is specified. 0: Cold reset of the Cortex-A55 is masked. 1: Cold reset of the Cortex-A55 is enabled.
7	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
6	WDTRSTSEL 6	0b	RW	Whether to mask the assertion of WDTOVFOUT due to a request from WDT channel 2 for the Cortex-M33 is specified. 0: Assertion of WDTOVFOUT is masked. 1: Assertion of WDTOVFOUT is enabled.
5	WDTRSTSEL 5	0b	RW	Whether to mask the assertion of WDTOVFOUT due to a request from WDT channel 1 for Cortex-A55 Core 1 is specified. 0: Assertion of WDTOVFOUT is masked. 1: Assertion of WDTOVFOUT is enabled.
4	WDTRSTSEL 4	0b	RW	Whether to mask the assertion of WDTOVFOUT due to a request from WDT channel 0 for Cortex-A55 Core 0 is specified. 0: Assertion of WDTOVFOUT is masked. 1: Assertion of WDTOVFOUT is enabled.
3	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
2	WDTRSTSEL 2	0b	RW	Whether to mask WDTSYSREST[2] due to a request from WDT channel 2 for the Cortex-M33 is specified. 0: System reset is masked. 1: System reset is enabled.
1	WDTRSTSEL 1	0b	RW	Whether to mask WDTSYSREST[1] due to a request from WDT channel 1 for Cortex-A55 Core 1 is specified. 0: System reset is masked. 1: System reset is enabled.

Bit	Bit Name	Initial Value	R/W	Description
0	WDTRSTSEL0	0b	RW	Whether to mask WDTSYSREST[0] due to a request from WDT channel 0 for Cortex-A55 Core 0 is specified. 0: System reset is masked. 1: System reset is enabled.

Note: Do not make the settings WDTRSTSEL3 = 1 and WDTRSTSEL7 = 0. If these settings are attempted, a system reset initializes the WDTRSTSEL7 bit to 1 (initial value) and WDTOVFOUT is unexpectedly asserted.

7.2.4.188 Cortex-A55 Cluster Power Status Monitor Register (CPG_CLUSTER_PCHMON)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is monitored by the CPU for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PDENY_MON	PACCEPT_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	PDENY_MON	—	R	Signal indicating the denial of power mode transition in the cluster
0	PACCEPT_MON	—	R	Signal indicating the acceptance of power mode transition in the cluster

7.2.4.189 Cortex-A55 Cluster Power Status Control Register (CPG_CLUSTER_PCHCTL)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is used for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PSTATE0_SET						
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PREQ_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 16	PSTATE0_SET	100_1000b	RW	Destination power mode of the cluster 100_1000b: ON 000_0000b: OFF
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PREQ_SET	0b	RW	Signal for indicating a request of power mode transition in the cluster

7.2.4.190 Cortex-A55 Core 0 Power Status Monitor Register (CPG_CORE0_PCHMON)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is monitored by the CPU for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PDENY0_MON	PACCEPT0_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	PDENY0_MON	—	R	Signal indicating the denial of power mode transition in the core 0
0	PACCEPT0_MON	—	R	Signal indicating the acceptance of power mode transition in the core 0

7.2.4.191 Cortex-A55 Core 0 Power Status Control Register (CPG_CORE0_PCHCTL)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is used for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	PSTATE0_SET					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PREQ0_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 16	PSTATE0_SET	00_1000b	RW	Destination power mode of the core 0 00_1000b: ON 00_0001b: OFF (Emulated) 00_0000b: OFF
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PREQ0_SET	0b	RW	Signal for indicating a request of power mode transition in the core 0

7.2.4.192 Cortex-A55 Core 1 Power Status Monitor Register (CPG_CORE1_PCHMON)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is monitored by the CPU for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PDENY1_MON	PACCEPT1_MON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	PDENY1_MON	—	R	Signal indicating the denial of power mode transition in the core 1
0	PACCEPT1_MON	—	R	Signal indicating the acceptance of power mode transition in the core 1

7.2.4.193 Cortex-A55 Core 1 Power Status Control Register (CPG_CORE1_PCHCTL)

A handshake through the P-Channel is necessary before a warm reset in the Cortex-A55.

This register is used for software control of the handshake.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	PSTATE1_SET					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PREQ1_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 16	PSTATE1_SET	00_0000b	RW	Destination power mode of the core 1 00_1000b: ON 00_0001b: OFF (Emulated) 00_0000b: OFF
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PREQ1_SET	0b	RW	Signal for indicating a request of power mode transition in the core 1

7.2.4.194 MSTOP Register ACPU (CPG_BUS_ACPU_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	MSTOP0_ON	0b	RW	The state of BUS_ACPU_MSTOP_MXSRAM_A operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.195 MSTOP Register MCPU1 (CPG_BUS_MCPU1_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTOP15_ON_WEN	MSTOP14_ON_WEN	MSTOP13_ON_WEN	MSTOP12_ON_WEN	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTOP15_ON	MSTOP14_ON	MSTOP13_ON	MSTOP12_ON	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MSTOP15_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	MSTOP14_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	MSTOP13_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	MSTOP12_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	MSTOP15_ON	0b	RW	The state of BUS_MCPU_MSTOP13_MHRSPI_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
14	MSTOP14_ON	0b	RW	The state of BUS_MCPU_MSTOP12_MHRSPI_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
13	MSTOP13_ON	0b	RW	The state of BUS_MCPU_MSTOP11_MHSSIF_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
12	MSTOP12_ON	0b	RW	The state of BUS_MCPU_MSTOP10_MHSSIF_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
11	MSTOP11_ON	0b	RW	The state of BUS_MCPU_MSTOP9_MHSSIF_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_MCPU_MSTOP8_MHSSIF_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_MCPU_MSTOP7_MHPOE3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_MCPU_MSTOP6_MHPOEGD operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
7	MSTOP7_ON	0b	RW	The state of BUS_MCPU_MSTOP5_MHPOEGC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_MCPU_MSTOP4_MHPOEGB operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_MCPU_MSTOP3_MHPOEGA operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_MCPU_MSTOP2_MHGPT operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_MCPU_MSTOP1_MHSRC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_MCPU_MSTOP0_MHMTU3A operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_MCPU_MSTOP_MHSPI operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_MCPU_MSTOP_MXSRAM_M operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.196 MSTOP Register MCPU2 (CPG_BUS_MCPU2_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTOP15_ON_WEN	MSTOP14_ON_WEN	MSTOP13_ON_WEN	MSTOP12_ON_WEN	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTOP15_ON	MSTOP14_ON	MSTOP13_ON	MSTOP12_ON	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MSTOP15_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	MSTOP14_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	MSTOP13_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	MSTOP12_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	MSTOP15_ON	0b	RW	The state of BUS_MCPU_MSTOP29_MPTSU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
14	MSTOP14_ON	0b	RW	The state of BUS_MCPU_MSTOP28_MPADC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
13	MSTOP13_ON	0b	RW	The state of BUS_MCPU_MSTOP27_MPI2C_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
12	MSTOP12_ON	0b	RW	The state of BUS_MCPU_MSTOP26_MPI2C_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
11	MSTOP11_ON	0b	RW	The state of BUS_MCPU_MSTOP25_MPI2C_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_MCPU_MSTOP24_MPI2C_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_MCPU_MSTOP23_MPCANFD operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_MCPU_MSTOP22_MHSCI_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
7	MSTOP7_ON	0b	RW	The state of BUS_MCPU_MSTOP21_MHSCI_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_MCPU_MSTOP20_MHIRDA operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_MCPU_MSTOP19_MHSCIF_4 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_MCPU_MSTOP18_MHSCIF_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_MCPU_MSTOP17_MHSCIF_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_MCPU_MSTOP16_MHSCIF_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_MCPU_MSTOP15_MHSCIF_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_MCPU_MSTOP14_MHRSPI_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.197 MSTOP Register PERI_COM (CPG_BUS_PERI_COM_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	MSTOP8_ON	0b	RW	The state of BUS_PERI_COM_MSTOP8_MXCOM operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
7	MSTOP7_ON	0b	RW	The state of BUS_PERI_COM_MSTOP7_MHUSB2_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_PERI_COM_MSTOP6_MHUSB2_0_F operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_PERI_COM_MSTOP5_MHUSB2_0_H operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_PERI_COM_MSTOP4_MPUSBT operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_PERI_COM_MSTOP3_MPGIGE_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_PERI_COM_MSTOP2_MPGIGE_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_COM_MSTOP1_MXSDHI_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_COM_MSTOP0_MXSDHI_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.198 MSTOP Register PERI_CPU (CPG_BUS_PERI_CPU_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTOP15_ON_WEN	MSTOP14_ON_WEN	MSTOP13_ON_WEN	MSTOP12_ON_WEN	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	—	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTOP15_ON	MSTOP14_ON	MSTOP13_ON	MSTOP12_ON	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	—	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MSTOP15_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP15_ON bit (bit 15) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
30	MSTOP14_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP14_ON bit (bit 14) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
29	MSTOP13_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP13_ON bit (bit 13) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
28	MSTOP12_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP12_ON bit (bit 12) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
27	MSTOP11_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP11_ON bit (bit 11) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15	MSTOP15_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP15_MXREG0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
14	MSTOP14_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP14_MPIM33 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
13	MSTOP13_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP13_MPIA55 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
12	MSTOP12_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP12_MPSRAM_M operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
11	MSTOP11_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP11_MPSRAM_A operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
10	MSTOP10_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP10_MPTZC_3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP9_MPTZC_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP8_MPTZC_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
7	MSTOP7_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP7_MPTZC_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP6_MHGPIO operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	MSTOP4_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP4_MPCPG operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP3_MPSYC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP2_MPCST operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP1_MXACPU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_CPU_MSTOP0_MXMCPU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.199 MSTOP Register PERI_DDR (CPG_BUS_PERI_DDR_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_DDR_MSTOP1_MXMEMC_REG operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_DDR_MSTOP0_MPPHY operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.200 MSTOP Register PERI_VIDEO (CPG_BUS_PERI_VIDEO_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	MSTOP10_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP10_ON bit (bit 10) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
25	MSTOP9_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24	MSTOP8_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP8_ON bit (bit 8) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	MSTOP10_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP10_MXVIDEO operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
9	MSTOP9_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP9_MPDU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
8	MSTOP8_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP8_MPFCPVD operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
7	MSTOP7_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP7_MPVSPD operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP6_MPDSIL operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP5_MPDSIPHY operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP4_MPISU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP3_MPCRU operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP2_MPFCPCS operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP1_MPVCP4L_C operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_PERI_VIDEO_MSTOP0_MPVCP4L_V operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.201 MSTOP Register REG0 (CPG_BUS_REG0_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	MSTOP6_ON	0b	RW	The state of BUS_REG0_MSTOP6_MPOSTM_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
5	MSTOP5_ON	0b	RW	The state of BUS_REG0_MSTOP5_MPOSTM_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_REG0_MSTOP4_MPOSTM_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_REG0_MSTOP3_MPWDT_1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_REG0_MSTOP2_MPWDT_0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_REG0_MSTOP1_MPWDT_2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

7.2.4.202 MSTOP Register REG1 (CPG_BUS_REG1_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	MSTOP7_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP7_ON bit (bit 7) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
22	MSTOP6_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP6_ON bit (bit 6) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
21	MSTOP5_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP5_ON bit (bit 5) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
20	MSTOP4_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP4_ON bit (bit 4) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7	MSTOP7_ON	0b	RW	The state of BUS_REG1_MSTOP7_MXGIC operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
6	MSTOP6_ON	0b	RW	The state of BUS_REG1_MSTOP6_MPTSIPG_OTP operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
5	MSTOP5_ON	0b	RW	The state of BUS_REG1_MSTOP5_MHTSIPG operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
4	MSTOP4_ON	0b	RW	The state of BUS_REG1_MSTOP4_MXMALI operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
3	MSTOP3_ON	0b	RW	The state of BUS_REG1_MSTOP3_MPDMAC_NS operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_REG1_MSTOP2_MXDMAC_NS operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_REG1_MSTOP1_MPDMAC_S operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_REG1_MSTOP0_MXDMAC_S operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.203 MSTOP Register TZCDDR (CPG_BUS_TZCDDR_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	MSTOP3_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	MSTOP2_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP2_ON bit (bit 2) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	MSTOP1_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP1_ON bit (bit 1) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	MSTOP3_ON	0b	RW	The state of BUS_TZCDDR_MSTOP3 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
2	MSTOP2_ON	0b	RW	The state of BUS_TZCDDR_MSTOP2 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
1	MSTOP1_ON	0b	RW	The state of BUS_TZCDDR_MSTOP1 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)
0	MSTOP0_ON	0b	RW	The state of BUS_TZCDDR_MSTOP0 operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.204 MSTOP Register MHU (CPG_MHU_MSTOP)

This register indicates the stop state of individual modules.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MHU_MSTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	R	R	R	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	MSTOP0_ON_WEN	0b	R0W1	Flag for enabling the writing to the MSTOP0_ON bit (bit 0) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	MHU_MSTOP	0b	RW	The state of MHU_MSTOP operation is indicated. 0: Normal operation 1: Module stop state (MSTP)

7.2.4.205 Other Function Register 1 (CPG_OTHERFUNC1_REG)

This register is used for miscellaneous functions.

Bit 0 is used to switch the source clock for the system bus and the IPs that can use SSCG clocks

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RES15_ON_WEN	RES14_ON_WEN	RES13_ON_WEN	RES12_ON_WEN	RES11_ON_WEN	RES10_ON_WEN	RES9_ON_WEN	RES8_ON_WEN	RES7_ON_WEN	RES6_ON_WEN	RES5_ON_WEN	RES4_ON_WEN	RES3_ON_WEN	RES2_ON_WEN	RES1_ON_WEN	RES0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES15_SET	RES14_SET	RES13_SET	RES12_SET	RES11_SET	RES10_SET	RES9_SET	RES8_SET	RES7_SET	RES6_SET	RES5_SET	RES4_SET	RES3_SET	RES2_SET	RES1_SET	RES0_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	RESn_ON_WEN [n=15 to 1]	All 0	R0W1	Flags for enabling the writing to the RESn_SET bits. These bits are always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	RES0_ON_WEN	0b	R0W1	Flag for enabling the writing to the RES0_SET bit (bit 0) (SEL_PLL5_4). This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	RESn_SET [n = 15 to 1]	All 0	RW	No function is assigned to these bits. When the flags for enabling the writing are set to 1, these bits can be written to. The written value is read from these bits.
0	RES0_SET	0b	RW	Selection of the clock to be supplied to SEL_PLL5_4 0: Up to 3,000MHz 1: 1500MHz

7.2.4.206 Other Function Register 2 (CPG_OTHERFUNC2_REG)

This register is used for miscellaneous functions.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RES15_ON_WEN	RES14_ON_WEN	RES13_ON_WEN	RES12_ON_WEN	RES11_ON_WEN	RES10_ON_WEN	RES9_ON_WEN	RES8_ON_WEN	RES7_ON_WEN	RES6_ON_WEN	RES5_ON_WEN	RES4_ON_WEN	RES3_ON_WEN	RES2_ON_WEN	RES1_ON_WEN	RES0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES15_SET	RES14_SET	RES13_SET	RES12_SET	RES11_SET	RES10_SET	RES9_SET	RES8_SET	RES7_SET	RES6_SET	RES5_SET	RES4_SET	RES3_SET	RES2_SET	RES1_SET	RES0_SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	RESn_ON_WEN [n=15 to 1]	All 0	R0W1	Flags for enabling the writing to the RESn_SET bits. These bits are always read as 0. 0: Writing is disabled. 1: Writing is enabled.
16	RES0_ON_WEN	0b	R0W1	Flag for enabling the writing to the RES0_SET bit (bit 0) (DDR_RST_N). This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
15 to 1	RESn_SET [n = 15 to 1]	All 0	RW	No function is assigned to these bits. When the flags for enabling the writing are set to 1, these bits can be written to. The written value is read from these bits.
0	RES0_SET	0b	RW	The DDR_RST_N reset terminal is controlled. 0: Reset signal is applied (reset state). 1: Reset signal is stopped (released from the reset state).

Note: The change to the setting in this register is immediately reflected in the reset signal. Therefore, no register is provided to monitor the result of the change.

7.2.4.207 Clock Control Register MBUS (CPG_CLKON_MBUS)

This register is used to supply or stop clocks for individual modules.

Use this register only when switching SEL_PLL3_3. Reserved bit is prohibited from setting anything other than the initial value.The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CLK9_ON_WEN	—	—	—	—	—	CLK3_ON_WEN	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R0W1	R	R	R	R	R	R0W1	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_ON	—	—	—	—	—	CLK3_ON	—	—	—
Initial Value	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	RW	R	R	R	R	R	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25	CLK9_ONWEN	0b	R0W1	Flag for enabling the writing to the CLK9_ON bit (bit 9) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
24 to 20	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19	CLK3_ON_WEN	0b	R0W1	Flag for enabling the writing to the CLK3_ON bit (bit 3) This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18 to 10	—	H'001	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9	CLK9_ON	1b	RW	The MBUS_CLK9 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
8 to 4	—	All 1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	CLK3_ON	1b	RW	The MBUS_CLK3 clock is supplied or stopped. 0: Clock is stopped. 1: Clock is supplied.
2 to 0	—	111b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

7.2.4.208 Clock Monitor Register MBUS (CPG_CLKMON_MBUS)

This register is used to monitor the state of the clocks supplied to individual modules.

The clocks listed in this register are not listed in the separate clock list.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLK9_MON	—	—	—	—	—	CLK3_MON	—	—	—
Initial Value	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11 to 10	—	01b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9	CLK9_MON	1b	R	The state of the MBUS_CLK9 clock is monitored. 0: Clock is stopped. 1: Clock is supplied.
8 to 4	—	All 1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	CLK3_MON	1b	R	The state of the MBUS_CLK3 clock is monitored. 0: Clock is stopped. 1: Clock is supplied.
2 to 0	—	111b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

7.3 Functions of the CPG

This section describes the functions of the CPG.

7.3.1 Clock Control

The CPG has the following five types of clock control functions.

- PLL control, setting, and monitoring functions
- Clock frequency dividers and selectors
- Fixed frequency dividers
- Clock enabling function

7.3.1.1 PLL Control, Setting, and Monitoring Functions

This section describes the functions for controlling, setting, and monitoring the PLL (PLL1 to PLL6).

The PLL is controlled through the registers in the CPG.

The PLL control module counts the PLL lock wait time to generate the PLL lock signal and it is used to control the PLL output clocks by the CGC so that invalid clocks are not output before the PLL is locked.

The reset state and lock state of the PLL can be monitored through registers in the CPG.

The following shows the PLL control, setting, and monitoring functions.

- PLL operating mode control
- Output clock setting
- PLL reset and lock state monitoring

These functions are described in detail in the following sections.

(1) PLL Operating Mode Control

The CPG registers are used to specify the SSCG PLL operating mode and make SSCG settings.

The normal mode or standby mode can be selected as the operating mode.

The following can be set up through CPG registers.

- Operating mode setting (normal mode or standby mode)
- SSCG enable or disable setting (valid in the normal mode)

The following table shows the registers to be used.

For details of the settings, refer to the functional description of the target register.

Table 7.6 Registers for Controlling the PLL Operating Mode

Register Name	Abbreviation	Function
PLLn (SSCG) standby control registers	CPG_SAMPLLn_STBY (n = 1, 4, or 6) CPG_SIPLL5_STBY	Standby mode setting (reset state) and SSCG enable or disable setting

(2) Output Clock Setting

The CPG registers are used to specify the clocks output from the PLL.

The following values can be set up.

- Output clock frequency setting
- SSCG modulation value setting

Table 7.7 Registers for Setting PLL Output Clocks

Register Name	Abbreviation	Function
PLLn (SSCG) output clock setting register 1	CPG_SAMPLLn_CLK1	Frequency setting
PLLn (SSCG) output clock setting register 2	CPG_SAMPLLn_CLK2	Frequency setting and SSCG modulation value setting

Note: n = 1, 4, or 6

Table 7.8 Registers for Setting PLL Output Clocks 2

Register Name	Abbreviation	Function
PLLn (SSCG) output clock setting register 1	CPG_SIPLL5_CLK1	Frequency setting
PLLn (SSCG) output clock setting register 3	CPG_SIPLL5_CLK3	Frequency setting and SSCG modulation value setting
PLLn (SSCG) output clock setting register 4	CPG_SIPLL5_CLK4	Frequency setting
PLLn (SSCG) output clock setting register 5	CPG_SIPLL5_CLK5	SSCG modulation value setting

(3) PLL State Monitoring

The reset state and lock state of the PLL can be monitored through CPG registers.

The following table shows the registers to be used.

Table 7.9 Registers for Monitoring the PLL State

Register Name	Abbreviation	Function
PLLn monitor registers	CPG_SAMPLLn_MON (n = 1, 4, or 6) CPG_SIPLLn_MON (n = 2, 3, or 5)	PLL state monitoring

(4) SSCG Switching in PLL3

In PLL3, the SSCG function is enabled or disabled through the MD_CLKS signal from the SYSC.

The state of the MD_CLKS signal is determined when the LSI is released from the system reset (PRST#) state and the signal state does not change after the system is activated.

The SSCG enabled or disable state in PLL3 cannot be switched through register settings.

MD_CLKS = 0: SSCG is disabled.

MD_CLKS = 1: SSCG is enabled.

7.3.1.2 Clock Frequency Dividers and Selectors

The CPG uses several frequency dividers and selectors to provide appropriate clocks that meet the specifications of individual modules. The frequency dividers and selectors are classified into the dynamic switching type that can be switched without generating a glitch and the static switching type that should be switched after the clock for the target module is stopped because a glitch is generated if this type of divider or selector is switched while the clock is supplied.

The following frequency dividers and selectors are provided.

- Fixed frequency dividers
- Dynamic switching selectors
- Dynamic switching variable-frequency dividers
  (with a duty ratio of 50:50 when divided by an even number)
- Static switching selectors
- Static switching variable-frequency dividers
  (with a duty ratio of 50:50 when divided by an even number)

A specific procedure should be used to switch each frequency divider or selector. Refer to **Section 7.4.6, Procedure for Switching the Division Ratio of the Dynamic Switching Frequency Dividers**, **Section 7.4.7, Procedure for Switching Clocks by the Dynamic Switching Frequency Selectors**, and **Section 7.4.8, Procedure for Switching Clocks by the Static Switching Frequency Dividers and Selectors (Procedure for Switching Source Clocks)**.

7.3.1.3 Fixed Frequency Dividers

A fixed frequency divider uses a fixed division value. The CPG provides 1/2, 1/4, 1/8, 1/10, 1/16, and 1/32 dividers, and all fixed dividers generate clocks with a duty ratio of 50:50.

7.3.1.4 PLL Clock Monitoring

This function monitors whether the PLL1 to PLL6 output clocks are operating.

The frequency in the CPG is divided to obtain the frequency that can be monitored through the external pin of this LSI and then output to the upper layer.

The following table shows the instance names and clock frequencies of the monitor points.

Table 7.10 Instance Names and Clock Frequencies of the Monitor Points

PLL Name	Instance Name of Monitor Point	Maximum Frequency (MHz)	Divider Value	Output Clock Frequency (MHz)
PLL1	CPG_CTL/PLL1/FOUT	1200	32	38
PLL2	CPG_CTL/PLL2/FOUT1PH0	1600 (1/2 of 3200)	32	50
PLL2	CPG_CTL/PLL2/FOUT3	533 (1/6 of 3200)	16	34
PLL3	CPG_CTL/PLL3/FOUT1PH0	1600 (1/2 of 3200)	32	50
PLL3	CPG_CTL/PLL3/FOUT3	533 (1/6 of 3200)	16	34
PLL3	CPG_CTL/PLL3/FOUT4	400 (1/8 of 3200)	8	50
PLL4	CPG_CTL/PLL4/FOUT	1600	32	50
PLL5	CPG_CTL/PLL5/FOUTPOSTDIV	3000	64	47
PLL5	CPG_CTL/PLL5/FOUT3	500 (1/6 of 3000)	16	32
PLL5	CPG_CTL/PLL5/FOUT1PH0	1500 (1/2 of 3000)	32	47
PLL6	CPG_CTL/PLL6/FOUT	500	16	32

7.3.2 Clock Generation and Control Functions

The following types of reset are provided.

Table 7.11 Reset Types

No.	Type	Method	Source	Range of Reset
1	System reset	Hardware reset	Reset by PRST#	All
2		WDT system reset	WDT overflow, etc.	All (The WDTOVF register is excluded.)
3	Module reset	WDT system reset	WDT overflow, etc.	CPU corresponding to each WDT
4		Software reset	Control through CPG registers	individual on-chip modules

7.3.2.1 Range of Reset Application

The following shows the application (target) range of the reset generated by each reset source.

(1) Hardware sources (PRST#, TRST#, and DEBUGEN)

(2) WDT system reset sources

Table 7.12 Range of Reset Application (1/2)

		Reset Source				Range of Reset Application				
						Modules Related to Debugging		WDT		
	Operating Mode	Debug Reset	System Reset							
No.	DEBUGEN	TRST#	PRST#	WDT Reset	JTAG Interface	Cortex-A55 Debug Reset (nPDGRESET)	Coresight Reset	Reset for WDTOVF Register	All Modules*1	Description of Reset
1	0	0	0	0	Reset	Reset	Reset	Reset	Reset	User mode, Hardware system reset
2	0	0	0	1	Reset	Reset	Reset	Reset	Reset	User mode, Hardware system reset
3	0	0	1	0	Reset	Reset	Reset	Reset	Reset	User mode, WDT system reset
4	0	0	1	1	Reset	Released	Reset	Released	Released	User mode, Normal operation state
5	0	1	0	0	Setting prohibited	—	—	—	—	—
6	0	1	0	1	Setting prohibited	—	—	—	—	—
7	0	1	1	0	Setting prohibited	—	—	—	—	—
8	0	1	1	1	Setting prohibited	—	—	—	—	—
9	1	0	0	0	Reset	Reset	Reset	Reset	Reset	Debug mode, Hardware system reset
10	1	0	0	1	Reset	Reset	Reset	Reset	Reset	Debug mode, Hardware system reset
11	1	0	1	0	Reset*2	Reset*3	Reset*2	Released	Reset*2	Debug mode, WDT system reset
12	1	0	1	1	Reset	Released	Released	Released	Released	Debug mode, Normal operation state
13	1	1	0	0	Reset	Reset	Reset	Released	Reset	Debug mode, Hardware system reset

Table 7.12 Range of Reset Application (2/2)

		Reset Source				Range of Reset Application				
						Modules Related to Debugging		WDT		
No.	Operating Mode DEBUGEN	Debug Reset TRST#	System Reset PRST#	WDT Reset	JTAG Interface	Cortex-A55 Debug Reset (nPDGRESET)	Coresight Reset	Reset for WDTOVF Register	All Modules*1	Description of Reset
14	1	1	0	1	Reset	Reset	Reset	Reset	Reset	Debug mode, Hardware system reset
15	1	1	1	0	Reset*2	Reset*3	Reset*2	Released	Reset*2	Debug mode, WDT system reset
16	1	1	1	1	Released	Released	Released	Released	Released	Debug mode, Normal operation state

Note 1. Reset when the system reset is enabled by the WDTRSTSEL register.

Note 2. Reset when the Cortex-A55 reset is enabled by the WDTRSTSEL register.

7.3.2.2 System Reset (External Pin)

A system reset is applied when the reset signal from the external PRST# pin is asserted and it initializes the entire LSI.

7.3.2.3 WDT System Reset

This LSI has three WDT channels and the CPG issues a reset request when any of the following sources is generated.

WDT Channel	Target of Monitoring	Function
Channel 0	Cortex-A55 Core 0	Reset due to a parity error in Cortex-A55 Core 0
Channel 1	Cortex-A55 Core 1	Reset due to a parity error in Cortex-A55 Core 1
Channel 2	Cortex-M33	Reset due to the LOCK signal in Cortex-M33

(1) WDT System Reset Circuit

The reset requests from three WDT channels can be assigned to the desired signals shown below by using the masking function specified through the settings of the WDTRSTSEL register implemented in the CPG.

If a bit in the WDTRSTSEL register is set to 1, the corresponding signal is asserted when the WDT issues a reset request.

Target of Monitoring	Function
WDT system reset	The reset request signal WDTRST# is sent to the reset generation circuit to apply a system reset.
WDTOVF_PERROUT	The WDTOVF_PERR signal is asserted to notify the external devices of the application of a system reset.
Cortex-A55 cold reset	A cold reset request is issued when either WDT channel 0 or 1 detects an error.

Bit	Bit Name	Initial Value	R/W	Reset Source Module	Function
0	WDTRSTSEL0	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When the corresponding WDT issues a reset request due to an overflow, etc. 0: WDT system reset is masked. 1: WDT system reset is applied.
1	WDTRSTSEL1	0	R/W	WDT (channel 1) Cortex-A55 Core 1	
2	WDTRSTSEL2	0	R/W	WDT (channel 2) Cortex-M33	
3	Reserved	1	R/W	—	—
4	WDTRSTSEL4	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When the corresponding WDT issues a reset request due to an overflow, etc. 0: Assertion of WDTOVFOUT is masked. 1: WDTOVFOUT is asserted.
5	WDTRSTSEL5	0	R/W	WDT (channel 1) Cortex-A55 Core 1	
6	WDTRSTSEL6	0	R/W	WDT (channel 2) Cortex-M33	
7	Reserved	1	R/W	—	—
8	WDTRSTSEL8	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When either WDT channel 0 or 1 issues a reset request[*1,*2] 0: Cold reset of the Cortex-A55 is masked. 1: Cold reset of the Cortex-A55 is applied.
9	WDTRSTSEL9	0	R/W	WDT (channel 1) Cortex-A55 Core 1	
10	WDTRSTSEL10	0	R/W	WDT (channel 2) Cortex-M33	When WDT channel 2 issues a reset request[*3] 0: Cold reset of the Cortex-M33 is masked. 1: Cold reset of the Cortex-M33 is applied.

Note 1. When an error occurs in either Core 0 or 1 in the Cortex-A55, a cold reset is applied to the entire Cortex-A55.

Note 2. For the timing of the cold reset in the Cortex-A55, refer to **Section 7.3.2.3(4), Timing of Cortex-A55 Cold Reset due to the WDT Source**.

Note 3. For the timing of the cold reset in the Cortex-M33, refer to **Section 7.3.2.3(5), Timing of Cortex-M33 Cold Reset due to the WDT Source**.

(2) WDTOVF_PERROUT Signal

This is an external signal that notifies the system that the WDT has applied a WDT system reset to this LSI.

If another WDT system reset is requested before the output of this signal pulse is completed, it is ignored.

The GPIO settings of the WDTOVF_PERROUT# pin such as the driving capability are reset by the GPIO_RST_WDTOVFN signal for the CPG_WDTOVFRST[3:0] register bits.

The following shows the sequence of reset by the WDT.

(1) The WDT channel selected by the WDTRSTSEL[7:4] bits generates a reset request.

(2) The CPG applies a WDT system reset.

(3) 64 cycles of the 24-MHz clock are waited.
* This is the period until the necessary settings are reflected in the entire LSI.

(4) A low pulse is output from the WDTOVF_PERROUT pin for 64 cycles of the 24-MHz clock.

(5) 64 cycles of the 24-MHz clock are waited.

(6) The WDT system reset signal is deasserted and the boot sequence is performed.

(3) WDT Overflow System Reset Register (CPG_WDTOVF_RST)

After the release from the WDT system reset applied by the WDT system reset circuit in response to a reset request such as a WDT overflow, this register is used to identify the WDT channel that generated the reset request.

This register is not reset by the WDT system reset generated by the WDT.

If a WDT system reset is not applied according to the WDTRSTSEL register setting, the WDT is not reset and the source of the reset request can be identified by checking the interrupt from the WDT and the status register in the WDT.

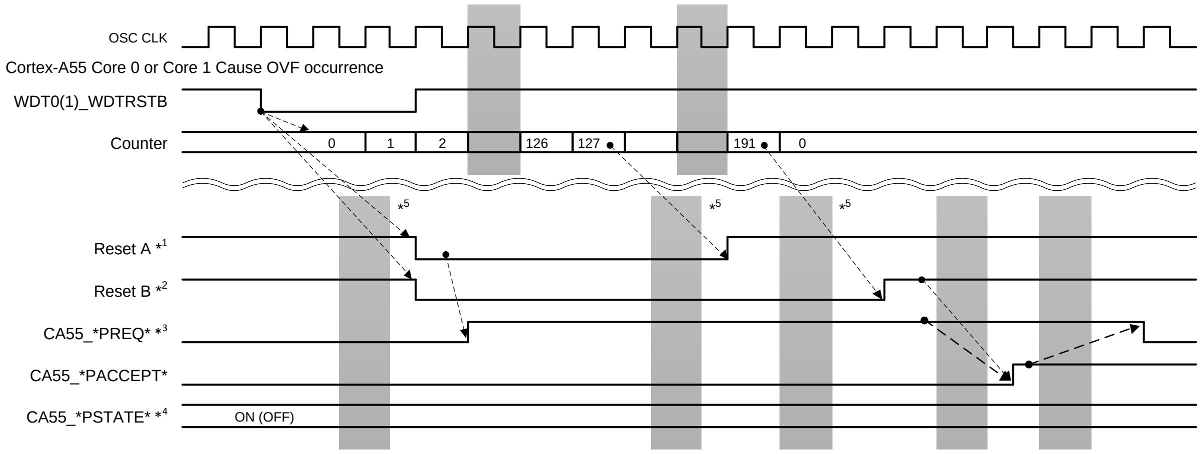
Table 7.13 WDT Overflow System Reset Register (CPG_WDTOVF_RST)

Bit	Bit Name	Initial Value	R/W	Reset Source Module	Function
0	WDTOVF0	0	R/W	WDT (channel 0) Cortex-A55 Core 0	When a WDT channel has applied a WDT system reset, the corresponding bit is set to 1. These bits are cleared by writing 1.
1	WDTOVF1	0	R/W	WDT (channel 1) Cortex-A55 Core 1	
2	WDTOVF2	0	R/W	WDT (channel 2) Cortex-M33	
15 to 3	Reserved	0	R	—	—
16	WDTOVF0_WEN	0	R0W1	—	Writing to the WDTOVF0 bit (bit 0) is masked. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
17	WDTOVF1_WEN	0	R0W1	—	Writing to the WDTOVF1 bit (bit 1) is masked. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
18	WDTOVF2_WEN	0	R0W1	—	Writing to the WDTOVF2 bit (bit 2) is masked. This bit is always read as 0. 0: Writing is disabled. 1: Writing is enabled.
31 to 19	Reserved	0	R	—	—

(4) Timing of Cortex-A55 Cold Reset due to the WDT Source

The following figure shows the timing of the cold reset for the Cortex-A55 alone due to the WDT source.

WDT Cortex-A55 Single Reset Time Chart



Note 1. CA55_RST10/CA55_RST11/CA55_RST12

Note 2. CA55_RST0[1:0]/CA55_RST3[1:0]/CA55_RST4/CA55_RST5/CA55_RST6/CA55_RST7/CA55_RST8/CA55_RST9

Note 3. The SYSREG setting must be set to ‘0’.

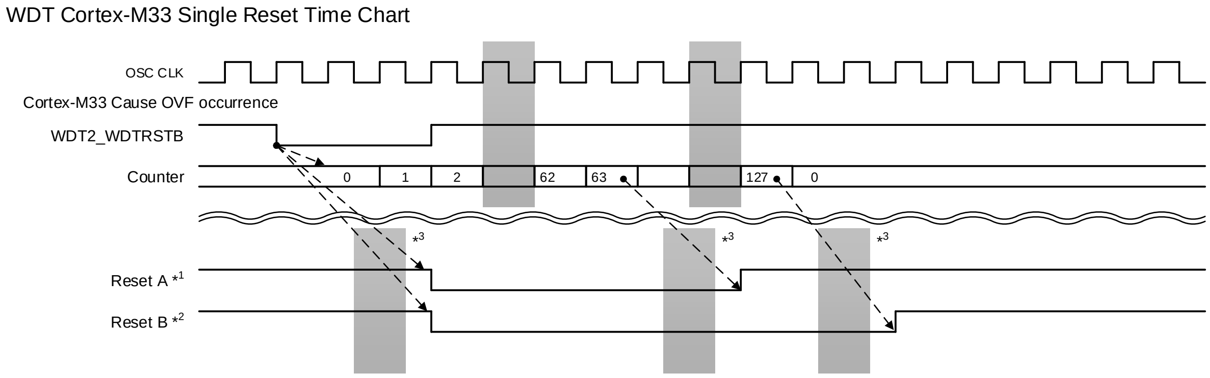
Note 4. The SYSREG setting is ‘ON’ for Core 0/Cluster and optional setting (on or off) for Core 1.

Note 5. Cortex-A55 reset generation has a certain time lag (approximately 35CLK@24MHz).

Figure 7.4 Timing of Cold Reset for the Cortex-A55 Alone due to the WDT Source

(5) Timing of Cortex-M33 Cold Reset due to the WDT Source

The following figure shows the timing of the cold reset for the Cortex-M33 alone due to the WDT source.



Note 1. CM33_MISCRESETN

Note 2. CA55_NPORESET/CM33_NSYSRESET

Note 3. Cortex-M33 reset generation has a certain time lag (approximately 35CLK@24MHz)

Figure 7.5 Timing of Cold Reset for the Cortex-M33 Alone due to the WDT Source

7.3.2.4 Cortex-A55 Reset

The following four types of reset corresponding to the target ranges of reset are provided for the Cortex-A55.

When the P-Channel circuit receives a request for a warm reset ((2) to (4) below), the P-Channel control is required before the application of a reset to the Cortex-A55.

(1) Cold reset
(2) Cluster warm reset
(3) Core 0 warm reset
(4) Core 1 warm reset

When a request to reset the Cortex-A55 alone is generated in the WDT, an error has probably occurred in the Cortex-A55, and the application of a warm reset may be difficult because the warm reset requires a handshake through the P-Channel. Therefore, a cold reset is applied to the Cortex-A55 instead of a warm reset.

- A warm reset for the Cortex-A55 is controlled by software executed in the Cortex-M33.
- After a cold reset (system reset) (normal), Cortex-A55 Core 0 is automatically activated by hardware.
- Refer to **Section 7.3.2.3(4), Timing of Cortex-A55 Cold Reset due to the WDT Source** for the operation of a cold reset (reset of a single core requested from the WDT). Reset control from CPG is performed by hardware.

(1) P-Channel Control for Cortex-A55 Reset

A handshake through the P-Channel is necessary before the application of a warm reset to the Cortex-A55. The registers in the CPG are monitored by the CPU for software control of the handshake. Three types of P-Channel are provided corresponding to the warm reset types. The CPG incorporates the registers (shown in the next page) for the following P-Channel signals and performs a handshake triggered by the PREQ signal. After the handshake, the reset signals for the target of the warm reset should be asserted and then negated by software to apply a warm reset.

Only the negation of the PREQ signal at the release of a cold reset (a system reset or a cold reset from the WDT) is controlled by the CPG hardware.

NOTES

1. The LSI that does not incorporate the Cortex-M33 does not support the warm reset of Cortex-A55.
2. When controlling the P-Channel by software, mask the cold reset of a single core of the Cortex-A55 from the WDT so that the cold reset is not applied. The system reset from the WDT does not need to be masked (the WDT system reset takes priority over the warm reset).
3. When enabling the cold reset of a single core of the Cortex-A55 from the WDT, make the “ON” setting through the PSTATE signal in the corresponding channel so that the PREQ signal is automatically controlled after the release from the reset state.

(a) P-Channel Control Registers in the Cortex-A55

The Cortex-A55 provides a P-Channel for power control for each of the regions — that is, the cluster region including the L3 and SCU, Core 0 region, and Core 1 region.

The P-Channel is driven in the PERIPHCLK domain, and the CPG contains the registers for controlling and monitoring the P-Channel signals other than the PACTIVE signal.

The CPG performs synchronization for the PACCEPT and PDENY signals.

For the PREQ and PSTATE signals, the Cortex-A55 has a synchronization circuit; the CPG does not perform synchronization.

For the P-Channel control registers, refer to following sections.

Section 7.2.4.188, Cortex-A55 Cluster Power Status Monitor Register (CPG_CLUSTER_PCHMON) to **Section 7.2.4.193, Cortex-A55 Core 1 Power Status Control Register (CPG_CORE1_PCHCTL)**

(b) P-Channel Handshake Control

The following describes the operation of the handshake through the P-Channel in the Cortex-A55.

Cold Reset (System Reset)

A handshake is necessary for a transition to the “ON” state.

Core 0: PREQ = 1 and PSTATE = ON in the initial state. After the release from the reset state, a handshake will be performed automatically and core 0 will be activated.
The CPG automatically clears PREQ for Core 0 to 0 after the release from the reset state.

Core 1: PREQ = 1 and PSTATE = OFF in the initial state. After the release from the reset state, a handshake will be performed automatically and core 1 will be stopped.
The CPG automatically clears PREQ for Core 1 to 0 after the release from the reset state.

Cold Reset (Cold Reset of the Cortex-A55 Alone Requested from the WDT)

A handshake is necessary for a transition to the “ON” state.

Core 0: When the cold reset of the Cortex-A55 alone requested from the WDT is enabled, set PREQ to 0 and PSTATE to "ON" in the register in advance. After the release from the reset state, a handshake is automatically executed to activate Core 0.
The CPG automatically clears PREQ for Core 0 to 0 after the release from the reset state.

Core 1: When the cold reset of the Cortex-A55 alone requested from the WDT is enabled, s Set PREQ to 0 and PSTATE to the desired value in advance in the register. After the release from the reset state, a handshake is automatically executed to activate Core 1 when PSTATE = “ON” or stop Core 1 when PSTATE = “OFF”.
The CPG automatically clears PREQ for Core 1 to 0 after the release from the reset state.

Warm Reset

Although the CPG has P-Channel control registers, it does not automatically control the P-Channel when a warm reset is requested; the user should control the P-Channel by software.

Figure 7.6 shows a flowchart and **Figure 7.7** shows a timing chart of the handshake for a warm reset through the Cortex-A55 P-Channel.

To execute the handshake, the user should manipulate registers in the CPG by software.

For a single-core Cortex-A55:

The PSTATE signal for Core 1 is fixed to the “OFF” state irrespective of the register settings when the Cortex-A55 incorporates a single core. Note that the PREQ signal for Core 1 is always controlled automatically regardless of whether the Cortex-A55 has a single-core or dual-core configuration.

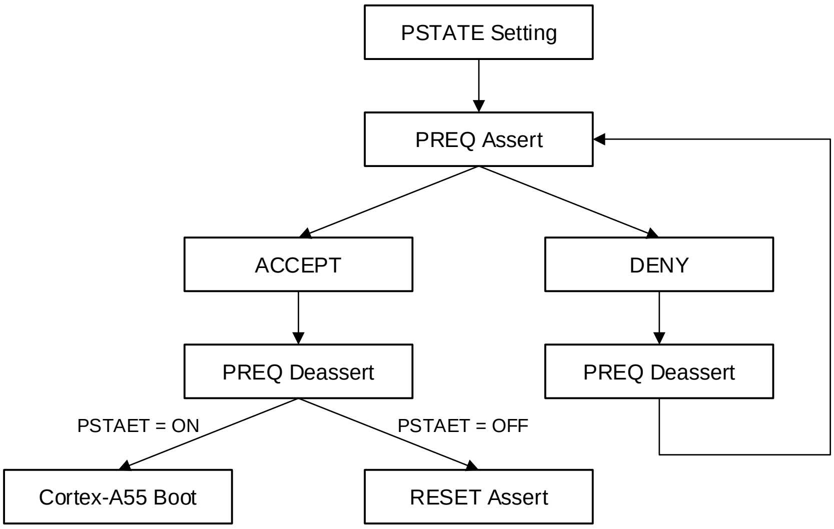
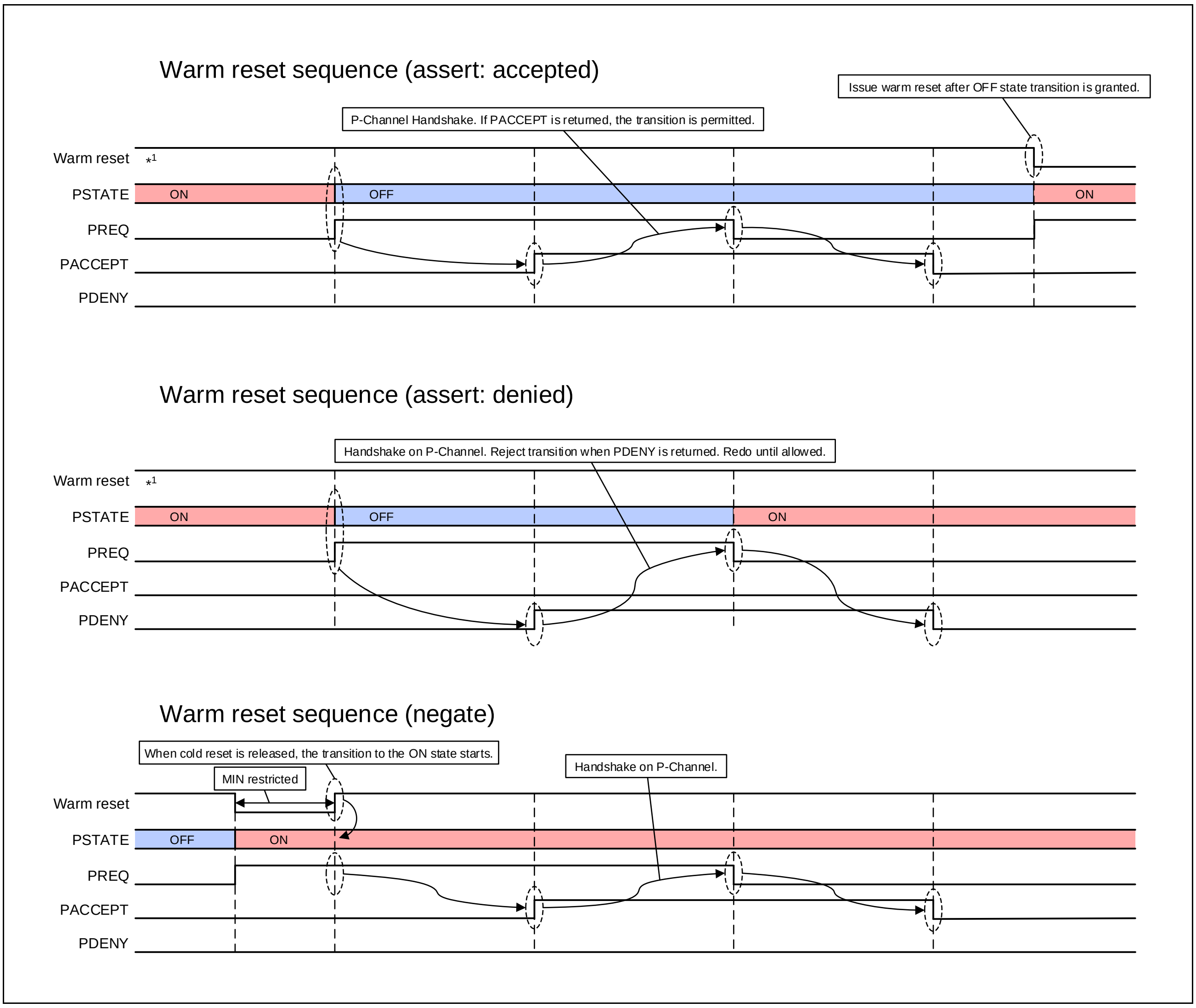


Figure 7.6 Flowchart of a Handshake through the Cortex-A55 P-Channel



Warm reset sequence (assert: accepted)

Issue warm reset after OFF state transition is granted.

P-Channel Handshake. If PACCEPT is returned, the transition is permitted.

Warm reset sequence (assert: denied)

Handshake on P-Channel. Reject transition when PDENY is returned. Redo until allowed.

Warm reset sequence (negate)

When cold reset is released, the transition to the ON state starts.

MIN restricted

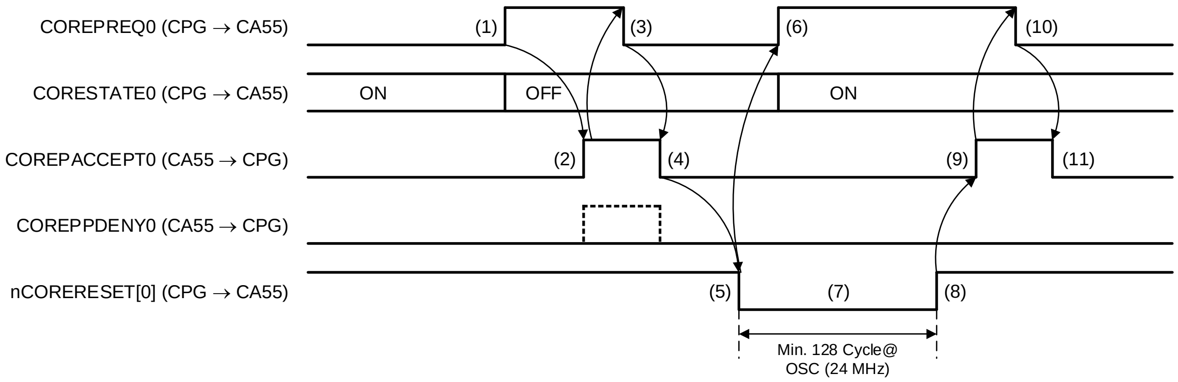
Handshake on P-Channel.

Figure 7.7 Timing Chart of a Handshake through the Cortex-A55 P-Channel (Warm Reset)

(c) Cortex-A55 Warm Reset Sequence

The following describes the warm reset sequence through the P-Channel.

Figure 7.8 shows a waveform when a warm reset of Cortex-A55 Core 0 is requested and accepted. When the request is denied, steps 1 to 4 are repeated until the request is accepted. The same sequence is used for the warm reset of the cluster or Core 1.



(1) The Cortex-M33 asserts COREPREQ0 according to CPG register settings (PSTATE = OFF).

(2) The Cortex-A55 asserts COREPACCEPT0.

(3) The Cortex-M33 deasserts COREPREQ0 according to CPG register settings.

(4) The Cortex-A55 deasserts COREPACCEPT0.

(5) The Cortex-M33 asserts the target reset signal (nCORERESET[0]) assigned to the warm reset of Cortex-A55 Core 0 according to CPG register settings.

(6) The Cortex-M33 asserts COREPREQ0 according to CPG register settings (PSTATE = ON).

(7) The Cortex-M33 waits for a certain period after the assertion of the reset signal.
18 cycles of the Cortex-A55 clock or 128 cycles at 24 MHz

(8) The Cortex-M33 deasserts the target reset signal (nCORERESET[0]) assigned to the warm reset of Cortex-A55 Core 0 according to CPG register settings.

(9) The Cortex-A55 asserts COREPACCEPT0.

(10) The Cortex-M33 deasserts COREPREQ0 according to CPG register settings.

(11) The Cortex-A55 deasserts COREPACCEPT0.

Figure 7.8 Cortex-A55 Warm Reset Sequence

(2) Operation of the Cortex-A55 Reset Control Circuit

The reset control circuit asserts the reset signal assigned to the reset request from the reset generator or software.

Reset Request Source	Reset Signal Assertion	Reset Signal Deassertion
System reset (including the reset from the WDT)	The CPG asserts the reset signal assigned to the cold reset.	The reset sequencer (boot sequencer) in the CPG releases the reset terminal assigned to the cold reset from the reset state. (The software reset register bit that corresponds to the reset terminal assigned to the cold reset indicates the reset released state in the initial state.)
Cold reset (software reset)*1	The Cortex-M33 asserts the reset signal assigned to the cold reset.	The Cortex-M33 deasserts the asserted signal so that the timing described in **Section 7.3.2.4(3), Timing of Cortex-A55 Reset**.
Cold reset (WDT)	The CPG asserts the reset signal assigned to the cold reset.	The CPG deasserts the asserted signal so that the timing described in **Section 7.3.2.3(4), Timing of Cortex-A55 Cold Reset due to the WDT Source**, is satisfied.
Cluster warm reset	The Cortex-M33 asserts the reset signal assigned to the cluster warm reset.	The Cortex-M33 deasserts the asserted signal after a period of 128 cycles at 24 MHz has elapsed.
Core 0 warm reset	The Cortex-M33 asserts the reset signal assigned to the Core 0 warm reset.	The Cortex-M33 deasserts the asserted signal after a period of 128 cycles at 24 MHz has elapsed.
Core 1 warm reset	The Cortex-M33 asserts the reset signal assigned to the Core 1 warm reset.	The Cortex-M33 deasserts the asserted signal after a period of 128 cycles at 24 MHz has elapsed.
Software reset*1	A software reset is applied by asserting the reset signal according to the data written to the register by software.	The CPG does not automatically deassert the reset signal. The reset signal retains the asserted state until the next access to the registers.

Note 1. The target reset signal should be controlled by software.

(3) Timing of Cortex-A55 Reset

There is a restriction on the timing of the release of the Cortex-A55 from the reset state.

When a reset is controlled by software, the restriction should be satisfied by software.

Hardware control is done for the system reset (including the reset from the WDT source) and the Cortex-A55 cold reset from the WDT.

The reset signals are classified into two groups:

Group (A):	nCPUPORESET[1:0], nCORERESET[1:0], nSPORESET, nSRESET, nPRESET, nATRESET, nGICRESET, and nPERIPHRESET
Group (B):	nARESET, nMISCRESET, and nPDBGRESET

To satisfy the restriction, the CPG deasserts group (B) after a period of 64 cycles at 24 MHz has elapsed since the deassertion of group (A).

7.3.2.5 Cortex-M33 Reset

Two types of reset corresponding to the target ranges of reset are provided for the Cortex-M33.

When a request to reset the Cortex-M33 alone is generated in the WDT, an error has probably occurred in the Cortex-M33, and the application of a warm reset may be difficult because the warm reset requires a handshake through the P-Channel. Therefore, a cold reset is applied to the Cortex-M33 instead of a warm reset.

- For the warm reset, refer to **Section 7.3.2.5(1), Handshake Control for Cortex-M33 Reset**, and **Section 7.3.2.5(2), Cortex-M33 Reset Sequence**. The reset control from the CPG is performed by hardware.
- To release the Cortex-M33 from the cold reset (system reset) state, refer to **Section 7.3.2.5(4), Timing of Cortex-M33 Reset**, and satisfy the described restriction. The reset control from the CPG needs software operation.
- The reset control from the CPG is performed by hardware.

MISCRESETn is a reset signal for the sub-system (SS) layer. This signal is used for a cold reset.
(For details, refer to the restriction on the timing of the Cortex-M33 cold reset.)

(1) Handshake Control for Cortex-M33 Reset

Like the warm reset for the Cortex-A55, the Cortex-M33 requires a handshake process to control the warm reset. After SYSRESETREQ is set to 1 in the Cortex-M33 and the Cortex-M33 enters the WFI state, the CPG executes a handshake with the Cortex-M33 to apply a warm reset to the Cortex-M33.

(2) Cortex-M33 Reset Sequence

For a warm reset for the Cortex-A55, the reset request signal nSYSRESET is issued as a result of the handshake between the CPG and Cortex-M33 and the signal pulse is generated for 128 cycles of the 24-MHz clock after the assertion of the signal.

The nSYSRESET signal is asserted under the conditions of SYSRESETREQ = high and SLEEPING = high.

(3) Operation of the Cortex-M33 Reset Control Circuit

The reset control circuit asserts the reset signal assigned to the reset request from the reset generator or software.

Reset Request Source	Reset Signal Assertion	Reset Signal Deassertion
System reset (including the reset from the WDT)	The software reset register is set to the initial value (initial value = cold reset).	The reset state is retained until the Cortex-A55 releases it.
Cold reset (software reset)*[1]	The Cortex-A55 asserts the following signals assigned to the cold reset. nPORESET nSYSRESET MISCRESTn	The Cortex-A55 deasserts MISCRESETn after a period of 64 cycles at 24 MHz has elapsed since the assertion of MISCRESETn. nSYSRESET and nPORESET are deasserted after a period of 64 cycles at 24 MHz has elapsed since the deassertion of MISCRESETn.
Cold reset (WDT)	The CPG asserts the following signals assigned to the cold reset. nPORESET nSYSRESET MISCRESTn	The CPG deasserts MISCRESETn after a period of 64 cycles at 24 MHz has elapsed since the assertion of MISCRESETn. nSYSRESET and nPORESET are deasserted after a period of 128 cycles at 24 MHz has elapsed since the deassertion of MISCRESETn. (See **Figure 7.5, Timing of Cold Reset for the Cortex-M33 Alone due to the WDT Source**.)
Warm reset	The CPG asserts the nSYSRESET signal assigned to the Cortex-M33 warm reset according to the procedure of Cortex-M33 warm reset.	The CPG deasserts the asserted signal after a period of 128 cycles at 24 MHz has elapsed.
Software reset*[1]	A software reset is applied by asserting the reset signal according to the data written to the register by software.	The reset signal is not deasserted. The reset signal retains the asserted state until the next access to the registers.

Note 1. The target reset signal should be controlled by software.

(4) Timing of Cortex-M33 Reset

There is a restriction on the timing of the release of the Cortex-M33 from the reset state.

When a reset is controlled by software, the restriction should be satisfied by software.

Hardware control is done for the Cortex-M33 cold reset from the WDT.

The reset signals are classified into two groups:

Group (A): MISCRESETn

Group (B): nPORESET and nSYSRESET

To satisfy the restriction, the CPG deasserts group (B) after a period of 64 cycles at 24 MHz has elapsed since the deassertion of group (A).

7.3.2.6 Module Reset Application Control

The application of a reset is controlled for individual modules.

In addition to a reset applied to individual modules according to register settings by software, some modules are reset in a certain reset mode.

For register setting specifications, refer to the descriptions of the reset control register *** (CPG_RST_***).

For a clock that is controlled by the corresponding CGC and is synchronized with a reset signal, the processing for stopping the clock is performed at the release of the reset state.

7.4 Operating Procedures

This section describes the procedures for operating the CPG.

7.4.1 Procedures for Supplying and Stopping Module Clocks

Use the following procedures to switch between the supply and stop of the target clock for a module.

* Sample procedures for the SRAM_MCPU clock (SRAM_MCPU_ACLK) are described here.

Supplying a Clock:

1. Setting the register for supplying the SRAM_MCPU clock (SRAM_MCPU_ACLK)

   Set up the following bits in the clock control register SRAM_MCPU (CPG_CLKON_SRAM_MCPU).
   — Setting for supplying the clock: Bit 0 (CLK0_ON) = 1 and bit 16 (CLK0_ONWEN) = 1

2. Confirming that the output of the SRAM_MCPU clock (SRAM_MCPU_ACLK) has started

   Read the following bit in the clock monitor register SRAM_MCPU (CPG_CLKMON_SRAM_MCPU) to confirm that the output of the clock has started.
   — Clock state: Bit 0 (UNIT0_CLK_MON), 1: Clock is supplied. (0: Clock is stopped.)

Stopping a Clock:

1. Setting the register for stopping the SRAM_MCPU clock (SRAM_MCPU_ACLK)

   Set up the following bits in the clock control register SRAM_MCPU (CPG_CLKON_SRAM_MCPU).
   — Setting for stopping the clock: Bit 0 (CLK0_ON) = 0 and bit 16 (CLK0_ONWEN) = 1

2. Confirming that the output of the SRAM_MCPU clock (SRAM_MCPU_ACLK) has been stopped

   Read the following bit in the clock monitor register SRAM_MCPU (CPG_CLKMON_SRAM_MCPU) to confirm that the output of the clock has been stopped.
   — Clock state: Bit 0 (UNIT0_CLK_MON), 0: Clock is stopped. (1: Clock is supplied.)

The following registers are used in the above procedures.

— 7.2.4.x Clock control registers ***** (CPG_CLKON_*****)

— 7.2.4.x Clock monitor registers ***** (CPG_CLKMON_*****)

For the procedures for supplying and stopping a clock signal, refer also to the section on the target module. If specific procedures for the module are described, use them.

7.4.2 Procedures for Supplying and Stopping Reset Signals

Use the following procedures to switch between the supply (reset state) and stop (released from the reset state) of the target reset signal for a module.

* Sample procedures for the SRAM_MCPU reset (SRAM_MCPU_ARESETN) are described here.

Supplying a Reset Signal:

1. Setting the register for supplying the SRAM_MCPU reset signal (SRAM_MCPU_ARESETN)

   Set up the following bits in the reset control register SRAM_MCPU (CPG_RST_SRAM_MCPU).
   — Setting for supplying the reset signal: Bit 0 (UNIT0_RSTB) = 1 and bit 16 (UNIT0_RST WEN) = 1

Stopping a Reset Signal:

1. Setting the register for stopping the SRAM_MCPU reset signal (SRAM_MCPU_ARESETN)

   Set up the following bits in the reset control register SRAM_MCPU (CPG_RST_SRAM_MCPU).

   - Setting for stopping the reset signal: Bit 0 (UNIT0_RSTB) = 0 and bit 16 (UNIT0_RST WEN) = 1
2. Confirming that the output of the SRAM_MCPU reset signal (SRAM_MCPU_ARESETN) has been stopped

   Read the following bit in the reset monitor register SRAM_MCPU (CPG_RSTMON_SRAM_MCPU) to confirm that the output of the reset signal has been stopped.
   — Reset state: Bit 0 (UNIT0_RST_MON)
   0: Reset signal is stopped (released from the reset state)
   1: Reset signal is output.

The following registers are used in the above procedures.

— 7.2.4.x Reset control registers ***** (CPG_RST_*****)

— 7.2.4.x Reset monitor registers ***** (CPG_RSTMON_*****)

For the procedures for supplying and stopping a reset signal, refer also to the section on the target module. If specific procedures for the module are described, use them.

7.4.3 Procedure for Activating Modules

Use the following procedure to activate an inactive module after a system reset.

For the specific procedure for activating the target module, refer to the section on the module.

1. Set up the clock control register for the clock signal connected to the target module to start the supply of the clock. Note that the PLL for the clock should be started before the clock if the PLL is stopped.
2. Read the clock monitor register to confirm that the supply of the target clock has started.
3. Set up the reset control register for the reset signal connected to the target module to release the module from the reset state.
4. Read the reset monitor register to conform that the module has been released from the reset state.

7.4.4 Procedure for Activating the Modules Related to PLL5

The synchronizer is intentionally removed from some PLL5 clock paths. For the following modules that use PLL5 clocks, use the activation flow and software control procedures shown in **Figure 7.9** and described later to avoid malfunctions due to hazards.

Applicable modules

- DSI
- LCDC

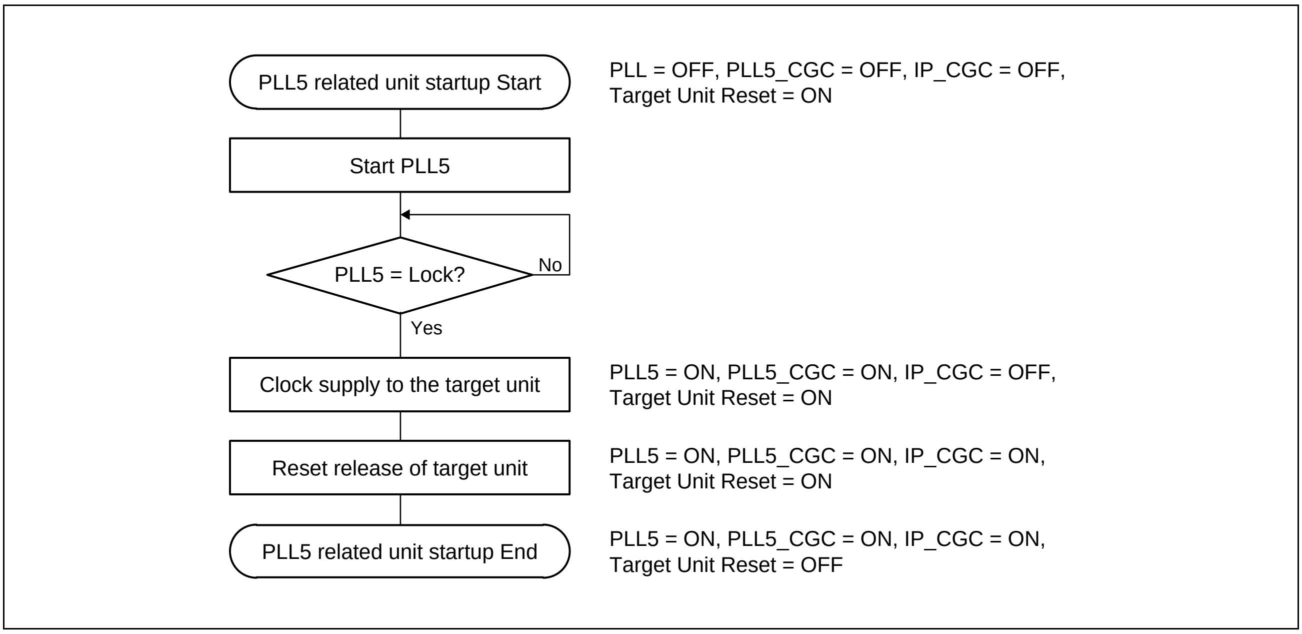


Figure 7.9 Flowchart of Activation of the Modules Related to PLL5

[Activation Flow]

<Initial state>

PLL5 = stopped, PLL5_CGC = activated, IP_CGC = activated,
and reset signals = applied (by hardware)

1) From the activation of PLL5 to the wait for the lock of PLL5:

PLL5 = activated (by software), PLL5_CGC = stopped, IP_CGC = stopped,
and reset signals for the target module = applied
In this period, the unstable clock before PLL5 is stopped by PLL5_CGC and does not propagate to the frequency divider or target module in the subsequent stages.

2) From the lock of PLL5 to the supply of the clock to the frequency divider DIV_DSI_A/B:

When PLL5 is locked, PLL5_CGC is automatically activated.

PLL5 = activated, PLL5_CGC = activated (by hardware), IP_CGC = stopped,
and reset signals for the applicable modules = applied

As soon as PLL5_CGC is activated, a clock hazard is momentarily present on the frequency divider but the incorrect output from the divider is stopped by the IP_CGC.
The frequency divider DIV_DSI_A will be stable in a short time.

The time until the divider becomes stable depends on the frequency setting in PLL5 but it is very short; there is no need to check whether it is stable by software.

3) Supplying clocks to the target module

Start the supply of the clocks to the target module by software.

PLL5 = activated, PLL5_CGC = activated, IP_CGC = activated (by software),
and reset signals for the target module = applied

Whether the IP_CGC is released from the reset state in synchronization with the clock differs between the PLLCLK_IN line and other lines.

For the PLLCLK_IN line, a synchronizer that synchronizes the maximum frequency (3 GHz) of PLLCLK_IN cannot be implemented by primitives. Therefore, the IP_CGC is released from the reset state asynchronously and a clock hazard will be present.
Accordingly, all reset signals for the target module (DSI) should be continued in this step.

4) Releasing the target module from the reset state

Release the target module from the reset state by software.

PLL5 = activated, PLL5_CGC = activated, IP_CGC = activated,
and reset signals = stopped (by software)

Before and after the release from the reset state, the clocks with which the reset signals are synchronized are stopped in the target module for several tens of clock cycles to avoid contention or adjust the timing flexibly.

Note that PLLCLK_IN does not stop because the DSI where PLLCLK_IN is supplied does not have reset signals synchronized with PLLCLK_IN.

7.4.5 Procedures for PLL Setup

The following shows the procedures for PLL setup.

7.4.5.1 Procedure for Setting PLL Normal Mode (Changing the Output Clock and SSCG Mode)

Use the following procedure to shift the PLL from the standby mode to the normal mode (change the output clock and SSCG mode).

[n = 1, 4, or 6: PLL1, PLL4, or PLL6]

1) Confirming the PLL operating mode (confirming that the PLL is in the standby mode)

- Check the following bit in the PLLn (SSCG) monitor register (CPG_SAMPLLn_MON).
- Shift to the standby mode: Confirm that bit 0 (PLLn_RESETB) = 0 (standby mode).
- Check the following bit in the PLLn (SSCG) monitor register (CPG_SAMPLLn_MON).
- Unlock of PLL: Confirm that bit 4 (PLLn_LOCK) = 0 (PLL is not locked).

Wait until the above conditions are satisfied.

2) Setting the output clock 1

- Modify the following bits in PLLn (SSCG) output clock setting register 1 (CPG_SAMPLLn_CLK1).
- For details of the settings, refer to **Section 7.2.4, Register Descriptions**.
- Setting of output clock (desired setting): Bits 5 to 0 (DIV_P_5 to DIV_P_0), bits 15 to 6 (DIV_M_9 to DIV_M_0), and bits 31 to 16 (DIV_K_15 to DIV_K_0)

3) Setting the output clock and SSCG modulation value 2

- Modify the following bits in PLLn (SSCG) output clock setting register 2 (CPG_SAMPLLn_CLK2).
- For details of the settings, refer to **Section 7.2.4, Register Descriptions**.
- Setting of output clock (desired setting): Bits 2 to 0 (DIV_S_2 to DIV_S_0)
- Setting of SSCG modulation value (desired setting): Bits 13 to 8 (MRR_5 to MRR_0) and bits 23 to 16 (MFR_7 to MFR_0)

4) Setting the PLL normal mode

- Set up the following bits in the PLLn (SSCG) standby control register (CPG_SAMPLLn_STBY).
- Setting of operating mode: Bit 0 (RESETB) = 1 (normal mode)
- Setting of SSCG mode: Bit 2 (SSCG_EN) = Desired setting

5) Confirming the shift to the PLL normal mode and the stable output of the clock

- Check the following bit in the PLLn (SSCG) monitor register (CPG_SAMPLLn_MON).
- Shift to standby mode: Confirm that bit 0 (PLLn_RESETB) = 1 (normal mode).
- Check the following bit in the PLLn (SSCG) monitor register (CPG_SAMPLLn_MON).
- Lock of PLL: Confirm that bit 4 (PLLn_LOCK) = 1 (PLL is locked).

Wait until the above conditions are satisfied.

NOTE

If there is no change in a parameter in step (2) or (3), the corresponding register does not need to be modified.

[PLL5]

1) PLL operation mode status check (confirmation that it is in standby mode)

- Check the following bits of the PLL5 (SSCG) monitor register (CPG_SIPLL5_MON).
- Standby mode transition confirmation: bit0 (PLLn_RESETB) = 0 (standby mode).
- Check the following bits of the PLL5 (SSCG) monitor register (CPG_SIPLL5_MON).
- Confirmation of PLL lock release: bit4 (PLLn_LOCK) = 0 (PLL is not locked).

Wait until the above conditions are satisfied.

2) Output clock setting 1

- Change the following bits of PLL5 (SSCG) output clock setting register 1 (CPG_SIPLL5_CLK1).
- For details on the setting values, refer to **Section 7.2.4, Register Descriptions**.
- Output frequency setting (optional):bit2-0(POSTDIV1)、bit6-4(POSTDIV2)、bit13-8(REFDIV)

3) Output clock setting, SSCG modulation value setting 3

- Change the following bits of PLLn (SSCG) output clock setting register 3 (CPG_SIPLL5_CLK3).
- For details of the setting values, refer to **Section 7.2.4, Register Descriptions**.
- Output frequency setting (optional): bit31-8(FRACIN)
- SSCG modulation value setting (optional): bit5-0(DIVVAL)

4) Output clock setting 4

- Change the following bits of PLL5 (SSCG) output clock setting register 4 (CPG_SIPLL5_CLK4).
- For details of the setting values, refer to **Section 7.2.4, Register Descriptions**.
- Output frequency setting (optional): bit27-16(INTIN)

5) SSCG modulation value setting 5

- Change the following bits of PLL5 (SSCG) output clock setting register 5 (CPG_SIPLL5_CLK5).
- For details of the setting values, refer to **Section 7.2.4, Register Descriptions**.
- SSCG modulation value setting (optional): bit4-0(SPREAD)

6) PLL normal mode setting

- Set the following bits of the PLLn (SSCG) standby control register (CPG_SIPLL5_STBY).
- Operation mode setting: bit0 (RESETB) = 1 (normal mode)
- SSCG mode setting: bit2 (SSC_EN) = optional

7) PLL normal mode transition, output clock stability check

- Check the following bits of the PLL5 (SSCG) monitor register (CPG_SIPLL5_MON).

- Confirmation of transition to normal mode: bit0 (PLLn_RESETB) = 1 (normal mode).
- Check the following bits of the PLL5 (SSCG) monitor register (CPG_SIPLL5_MON).
- PLL Lock confirmation: Bit4 (PLLn_LOCK) = 1 (PLL Lock).

Wait until the above conditions are satisfied.

NOTE

If there is no change in a parameter in step (2), (3), (4) or (5), the corresponding register does not need to be modified.

7.4.5.2 Procedure for Setting PLL Standby Mode

Use the following procedure to shift the PLL from the normal standby mode to the standby mode.

[n = 1, 4, or 6: PLL1, PLL4, or PLL6]

1) Setting the PLL standby mode

- Set the following bits of the PLL5 (SSCG) standby control register (CPG_SAM / SIPLL5_STBY).
- Operation mode setting: bit0 (RESETB) = 0 (standby mode) (other bits are “0”).

2) Confirming the shift to the PLL standby mode

- Check the following bits of the PLLn (SSCG) monitor register (CPG_SAM / SIPLL5_MON).
- Standby mode transition confirmation: bit0 (PLLn_RESETB) = 0 (standby mode).
- Check the following bits of the PLLn (SSCG) monitor register (CPG_SAM / SIPLL5_MON).
- Confirmation of PLL lock release: bit4 (PLLn_LOCK) = 0 (PLL is not locked).

Wait until the above conditions are satisfied.

7.4.5.3 PLLs with Restrictions

PLL1 to PLL3 cannot be switched to desired clock settings by the user because these PLLs output clocks for the CPU and bus operation.

PLL1 outputs clocks dedicated to the Cortex-A55. To modify its settings after booting, the registers should be manipulated from the Cortex-M33.

The PLL2 and PLL3 settings cannot be modified after booting because these PLLs output system clocks for bus operation.

PLL4 to PLL6 can be set up, started, or stopped by the user at the desired timing under the allowable conditions of use for the target module.

7.4.6 Procedure for Switching the Division Ratio of the Dynamic Switching Frequency Dividers

Use the following procedure to set the division ratio of the dynamic switching frequency dividers.

* A sample procedure for DIV_PLL2_A is described here.

1) Confirming the DIV_PLL2_A state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKSTATUS).
- DIV_PLL2_A state: Confirm that bit 0 (DIVPL2A_STS) = 0 (switching is completed).

Wait until the above condition is satisfied.

2) Setting the DIV_PLL2_A division ratio

- Set up the following bits in the division ratio setting (PLL2) register (CPG_PL2_DDIV).
- Setting of division ratio: Bits 1 and 0 (DIVPL2A_SET) = desired value and bit 16 (DIV_PLL2_A_WEN) = 1

3) Confirming the DIV_PLL2_A state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKSTATUS).
- DIV_PLL2_A state: Confirm that bit 0 (DIVPL2A_STS) = 0 (switching is completed).

Wait until the above condition is satisfied.

The following registers are used in the above procedure.

— Division ratio setting (PLL1) register (CPG_PL1_DDIV)
— Division ratio setting (PLL2) register (CPG_PL2_DDIV)
— Division ratio setting (PLL3) register (CPG_PL3A_DDIV)
— Division ratio setting (PLL3) register (CPG_PL3B_DDIV)
— Division ratio setting (PLL6) register (CPG_PL6_DDIV)

The following status monitor register is used.

— Clock status monitor register (CPG_CLKSTATUS)

When 1 is written to a xx_WEN bit placed in the upper 16 bits of a division ratio setting register, the clock switching control begins even if the settings in the lower bits are not changed. Check the status monitor register (CPG_CLKSTATUS) and wait until the switching is completed. The clock will temporarily stop at the timing of switching.

7.4.7 Procedure for Switching Clocks by the Dynamic Switching Frequency Selectors

Use the following procedure to switch clocks by the dynamic switching frequency selectors.

* A sample procedure for SEL_PLL4 is described here.

1) Confirming the SEL_PLL4 state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKSTATUS).
- SEL_PLL4 state: Confirm that bit 31 (SELPLL4_STS) = 0 (switching is completed).

Wait until the above condition is satisfied.

2) Setting the switching of the clock output from SEL_PLL4

- Set up the following bits in the source clock setting (DDR) register (CPG_PL4_DSEL).
- Setting of clocks switching (switch from PLL4 to OSC_DIV1000): Bits 31 to 0 = H’0001_0000
  (switch from OSC_DIV1000 to PLL4): Bits 31 to 0 = H’0001_0001

Confirming the SEL_PLL4 state (checking that it is not busy)

- Check the following bit in the clock status monitor register (CPG_CLKSTATUS).
- SEL_PLL4 state: Confirm that bit 31 (SELPLL4_STS) = 0 (switching is completed).

Wait until the above condition is satisfied.

The following registers are used in the above procedure.

— Source clock setting (DDR) register (CPG_PL4_DSEL)

— Source clock setting (SDHI) register (CPG_PL2SDHI_DSEL)

When 1 is written to a xx_WEN bit placed in the upper 16 bits of a source clock setting register, the clock switching control begins even if the settings in the lower bits are not changed. Check the status monitor register (CPG_CLKSTATUS) and wait until the switching is completed. The clock will temporarily stop at the timing of switching.

7.4.8 Procedure for Switching Clocks by the Static Switching Frequency Dividers and Selectors (Procedure for Switching Source Clocks)

Use the following procedure to set the division ratio of the static switching frequency dividers and to switch clocks by the static switching frequency selectors.

When a source clock is active: Stop the clock. → Make settings for switching clocks. → Start the clock.

When a source clock is stopped: Make settings for switching.

The following registers are used in the above procedure.

— Source clock setting register (CPG_PL3_SSEL)*[1]

— Source clock setting register (CPG_PL6_SSEL)

— Source clock setting register (CPG_PL6_ETH_SSEL)*[2]

— Division ratio setting (PLL5) register (CPG_PL5_SDIV)

— Bit 0 (RES0_SET) in other function register 1 (CPG_OTHERFUNC1_REG)

Note 1. When switching SEL_PLL3_3 in CPG_PL3_SSEL, please note the following points.

1) Do not access SPI-Multi during switching.
2) If a TZC2INT interrupt is occurring, allow the interrupt to complete before stopping the clock on the CPG_CLKON_MBUS.
3) Before switching, stop the following clocks.
    - CPG_CLKON_SPI_MULTI (bit1, bit0)
    - CPG_CLKON_MBUS (bit9, bit3)
4) CPG_CLKON_SPI_MULTI and CPG_CLKON_MBUS are not in the order to stop the clock.

Note 2. Set the CPG_PL6_ETH_SSEL in the ETHER module stopped state (reset state).

7.4.9 Boot Sequence (Normal Operation)

After the reset sequence ends, perform the following operations if necessary.

- Start PLLs.
- Set up clock selectors.
- Set up clock division ratios.
- Supply clocks to modules.
- Release modules from the reset state.

7.4.10 Boot Sequence (Debug Mode)

After the reset sequence (debug mode) ends, perform the following operations if necessary.

- Start PLLs.
- Set up clock selectors.
- Set up clock division ratios.
- Supply clocks to modules.
- Release modules from the reset state.

7.4.11 General Operating Procedures

The following describes general operating procedures.

7.4.11.1 Procedure for Writing to Registers

The registers in the CPG have write-enable flags in the upper 16 bits and normal data fields in the lower 16 bits except for the registers shown below. Therefore, only the desired bit can be modified without read-modify-write operation.

To modify the value of a bit, write 1 to the write-enable flag having the target bit number + 16 and write a desired value to the target bit.

(Examples: To modify bit 0 to 1b, write H’0001_0001. To modify it to 0b, write H’0001_0000. When H’0000_xxxx is written, the value of bits 15 to 0 does not change.)

NOTE

In some registers, a single write-enable flag controls multiple data bit. Refer to the description of each register.

Registers that have data fields in the upper bits instead of write-enable flags:

— PLLn (SSCG) output clock setting register 1 (CPG_SAMPLLn_CLK1): (n = 1, 4, or 6)

— PLLn (SSCG) output clock setting register 2 (CPG_SAMPLLn_CLK2): (n = 1, 4, or 6)

— PLLn (SSCG) output clock setting register 3 (CPG_SIPLLn_CLK3): (n = 5)

— PLLn (SSCG) output clock setting register 4 (CPG_SIPLLn_CLK4): (n = 5)

— PLLn (SSCG) output clock setting register 5 (CPG_SIPLLn_CLK5): (n = 5)

— Cortex-A55 cluster power status control register (CPG_CLUSTER_PCHCTL)

— Cortex-A55 Core 0 power status control register (CPG_CORE0_PCHCTL)

— Cortex-A55 Core 1 power status control register (CPG_CORE1_PCHCTL)

8. Interrupt Controller

The interrupt controller decides the priority of interrupt sources and controls interrupt requests to the CPU.

The interrupt controller registers set the order of priority of each interrupt, allowing the user to process interrupt requests according to the user-set priority.

8.1 Features

■ **Equipped with Arm® CoreLink™ GIC-600 Generic Interrupt Controller*1 for Arm Cortex-A55.**

- 32 levels of priority.
- SGI (Software Generated Interrupt): 16 interrupts
- PPI (Private Peripheral Interrupt): 16 interrupts
- SPI (Shared Peripheral Interrupt): 480 interrupts
- LPI (Locality-specific Peripheral Interrupt): Not supported

■ **Nested Vectored Interrupt Controller (NVIC)*2 built in Arm Cortex-M33.**

- 256 levels of priority
- 480 interrupts

■ **External Interrupts**

- NMI, IRQ0-7, TINT0-31
- Noise filter function.*3

■ **Internal Interrupts**

- On-chip peripheral interrupts
- On-chip Bus error interrupts
- On-chip RAM ECC error interrupts

Note 1. Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual.
Note 2. Refer to Arm® Cortex®-M33 Processor Revision: r0p4 Technical Reference Manual.
Note 3. Refer to **Section 41, General Purpose Input Output Port (GPIO)**.

8.1.1 External Signal Pins

The following table shows external signal pins for Interrupt Controller.

Table 8.1 External Signal Pins

Name	Width	I/O	Description
NMI	1	I	NMI (Non Maskable Interrupt) pin. Refer to **Section 8.7.1**.
IRQ0 – IRQ7	8	I	IRQ (Interrupt Request) pins. GPIO pins can be used as IRQ pins. Refer to **Section 8.7.2**.
P0_0 – P48_4	123	I	GPIO pins. GPIO pins can be used as external interrupt input pins (GPIOINT0-122). Assign only 32 pins in GPIOINT0-122 to TINT0-31. Refer to **Section 8.7.3**.

8.1.2 Block Diagram

The block diagram of Interrupt Controller is as follows.

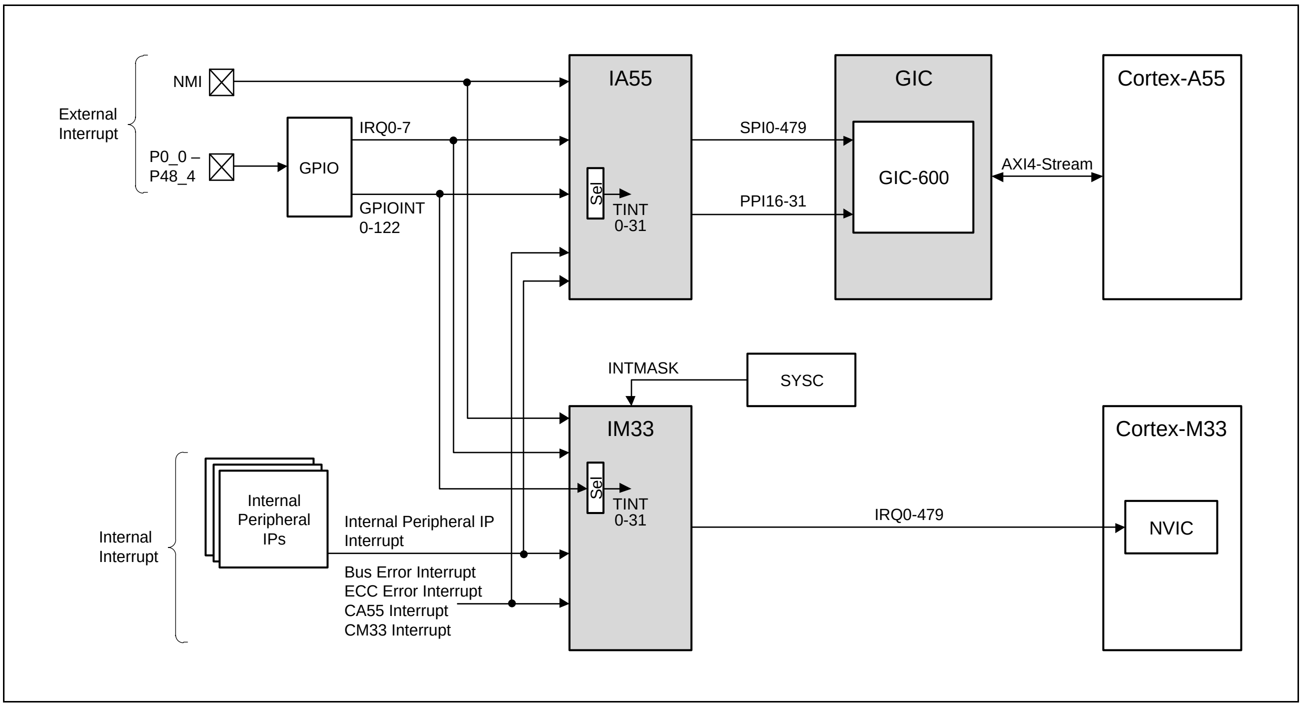


Figure 8.1 Interrupt Controller Block Diagram

GIC

GIC is equipped with Arm® CoreLink™ GIC-600 Generic Interrupt Controller for Cortex-A55.

IA55

IA55 performs various interrupt controls including synchronization for the external interrupts of NMI, IRQ, and GPIOINT and the interrupts of the built-in peripheral interrupts output by each IP. And it notifies the interrupt to the GIC.

- Select 32 TINT from 123 GPIOINT.
- Integration of bus error interrupts from system bus.
- Integration of ECC error interrupts from On-chip RAM.
- Indicate interrupt status. (NMI, IRQ, TINT, integrated bus error interrupt and integrated ECC error interrupt)
- Setting of interrupt detection method. (NMI, IRQ and TINT)

IM33

IM33 performs various interrupt controls including synchronization for the external interrupts of NMI, IRQ and GPIOINT and internal peripheral interrupts output by each IP. And it notifies the interrupt to the built-in interrupt controller (NVIC) for Cortex-M33.

- Select 32 TINT from 123 GPIOINT.
- Integration of bus error interrupts from system bus.
- Integration of ECC error interrupts from On-chip RAM.
- Indicate interrupt status. (NMI, IRQ, TINT, integrated bus error interrupt and integrated ECC error interrupt)

- Set interrupt detection method. (NMI, IRQ and TINT)
- All interrupts can be masked by setting of SYSC Register. (IM33_MASK bit of SYS_LP_CLT7 register.)
  Refer to **Section 42, Low Power Mode**.

8.2 Interrupt Mapping

The following table indicates interrupt mapping for Cortex-A55 and Cortex-M33.

Refer to **Section 14, Direct Memory Access Controller** with the interrupts through DMA Controller, and **Section 8.8.1**.

Table 8.2 Interrupt mapping (1/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
GIC (Software Interrupt)	SGI ID0	0	SGI 0	—	—	
	SGI ID1	1	SGI 1	—	—	
	SGI ID2	2	SGI 2	—	—	
	SGI ID3	3	SGI 3	—	—	
	SGI ID4	4	SGI 4	—	—	
	SGI ID5	5	SGI 5	—	—	
	SGI ID6	6	SGI 6	—	—	
	SGI ID7	7	SGI 7	—	—	
	SGI ID8	8	SGI 8	—	—	
	SGI ID9	9	SGI 9	—	—	
	SGI ID10	10	SGI 10	—	—	
	SGI ID11	11	SGI 11	—	—	
	SGI ID12	12	SGI 12	—	—	
	SGI ID13	13	SGI 13	—	—	
	SGI ID14	14	SGI 14	—	—	
	SGI ID15	15	SGI 15	—	—	
Reserved	—	16	PPI 0	—	—	
Reserved	—	17	PPI 1	—	—	
Reserved	—	18	PPI 2	—	—	
Reserved	—	19	PPI 3	—	—	
Reserved	—	20	PPI 4	—	—	
Reserved	—	21	PPI 5	—	—	
Cortex-A55	nCOMIRQ	22	PPI 6	—	Level	
	nPMUIRQ	23	PPI 7	—	Level	
	CTIIRQ	24	PPI 8	—	Level	
	nVCPUMNTIRQ	25	PPI 9	—	Level	
	nCNTHPIRQ	26	PPI 10	—	Level	
	nCNTVIRQ	27	PPI 11	—	Level	
	nCNTHVIRQ	28	PPI 12	—	Level	
	nCNTPSIRQ	29	PPI 13	—	Level	
	nCNTPNSIRQ	30	PPI 14	—	Level	
Reserved	—	31	PPI 15	—	—	
NMI pin	NMI	32	SPI 0	IRQ 0	Level	*4

Table 8.2 Interrupt mapping (2/13)

		Cortex-A55		Cortex-M33		
Interrupt Source*3	Cause of Interrupt	Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1	Interrupt Type*2	Note
IRQ pins	IRQ0	33	SPI 1	IRQ 1	Level	*5
	IRQ1	34	SPI 2	IRQ 2	Level	*5
	IRQ2	35	SPI 3	IRQ 3	Level	*5
	IRQ3	36	SPI 4	IRQ 4	Level	*5
	IRQ4	37	SPI 5	IRQ 5	Level	*5
	IRQ5	38	SPI 6	IRQ 6	Level	*5
	IRQ6	39	SPI 7	IRQ 7	Level	*5
	IRQ7	40	SPI 8	IRQ 8	Level	*5
Cortex-A55	nCOMMIRQ[1]	41	SPI 9	IRQ 9	Level	
	nCOMMIRQ[0]	42	SPI 10	IRQ 10	Level	
	nPMUIRQ[1]	43	SPI 11	IRQ 11	Level	
	nPMUIRQ[0]	44	SPI 12	IRQ 12	Level	
	nCLUSTERPMUIRQ	45	SPI 13	IRQ 13	Level	
	nERRIRQ[2]	46	SPI 14	IRQ 14	Level	
	nERRIRQ[1]	47	SPI 15	IRQ 15	Level	
	nERRIRQ[0]	48	SPI 16	IRQ 16	Level	
	nFAULTIRQ[2]	49	SPI 17	IRQ 17	Level	
	nFAULTIRQ[1]	50	SPI 18	IRQ 18	Level	
	nFAULTIRQ[0]	51	SPI 19	IRQ 19	Level	
Cortex-M33	CTIIRQ[1]	52	SPI 20	IRQ 20	Level	
	CTIIRQ[0]	53	SPI 21	IRQ 21	Level	
GIC	fault_int	54	SPI 22	IRQ 22	Level	
	err_int	55	SPI 23	IRQ 23	Level	
	pmu_int	56	SPI 24	IRQ 24	Level	
System Bus	BUS_ERR_INT	57	SPI 25	IRQ 25	Edge	*7
System Bus (TZC for SRAM ACPU)	TZC0INT	58	SPI 26	IRQ 26	Level	
System Bus (TZC for SRAM MCPU)	TZC1INT	59	SPI 27	IRQ 27	Level	
System Bus (TZC for SPI Multi)	TZC2INT	60	SPI 28	IRQ 28	Level	
System Bus (TZC for DDR MEMC)	TZC3INT	61	SPI 29	IRQ 29	Level	
System Bus (CM33 System bus)	ERRINT_S	62	SPI 30	IRQ 30	Edge	
System Bus (CM33 Code bus)	ERRINT_C	63	SPI 31	IRQ 31	Edge	
Reserved	—	64	SPI 32	IRQ 32	—	
Reserved	—	65	SPI 33	IRQ 33	—	
ECCRAM0	EC7TIE1_0	66	SPI 34	IRQ 34	Edge	*8
	EC7TIE2_0	67	SPI 35	IRQ 35	Edge	*8
	EC7TIOVF_0	68	SPI 36	IRQ 36	Edge	*8
ECCRAM1	EC7TIE1_1	69	SPI 37	IRQ 37	Edge	*8
	EC7TIE2_1	70	SPI 38	IRQ 38	Edge	*8
	EC7TIOVF_1	71	SPI 39	IRQ 39	Edge	*8
DDR3L/4	controller_int	72	SPI 40	IRQ 40	Level	

Table 8.2 Interrupt mapping (3/13)

		Cortex-A55		Cortex-M33		
Interrupt Source*3	Cause of Interrupt	Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1	Interrupt Type*2	Note
SPI Multi	QSPI_INT#	73	SPI 41	IRQ 41	Level	
SYSC	SYS_LPM_INT	74	SPI 42	IRQ 42	Level	
	SYS_CA55STBYDONE_INT	75	SPI 43	IRQ 43	Level	
	SYS_CM33STBYR_INT	76	SPI 44	IRQ 44	Level	
	SYS_CA55_DENY	77	SPI 45	IRQ 45	Level	
GTM (ch0)	OSTM0TINT	78	SPI 46	IRQ 46	Edge	
GTM (ch1)	OSTM1TINT	79	SPI 47	IRQ 47	Edge	
GTM (ch2)	OSTM2TINT	80	SPI 48	IRQ 48	Edge	
WDT (CA55 core0)	WDTINT_A0	81	SPI 49	IRQ 49	Level	
	PERROUT_A0	82	SPI 50	IRQ 50	Level	
WDT (CA55 core1)	WDTINT_A1	83	SPI 51	IRQ 51	Level	
	PERROUT_A1	84	SPI 52	IRQ 52	Level	
WDT (CM33)	WDTINT_M	85	SPI 53	IRQ 53	Level	
	PERROUT_M	86	SPI 54	IRQ 54	Level	
Reserved	—	87	SPI 55	IRQ 55	—	
MHU (secure)	msg_ch0_s	88	SPI 56	IRQ 56	Level	
	msg_ch1_s	89	SPI 57	IRQ 57	Level	
	msg_ch2_s	90	SPI 58	IRQ 58	Level	
	msg_ch3_s	91	SPI 59	IRQ 59	Level	
	msg_ch4_s	92	SPI 60	IRQ 60	Level	
	msg_ch5_s	93	SPI 61	IRQ 61	Level	
	rsp_ch0_s	94	SPI 62	IRQ 62	Level	
	rsp_ch1_s	95	SPI 63	IRQ 63	Level	
	rsp_ch2_s	96	SPI 64	IRQ 64	Level	
	rsp_ch3_s	97	SPI 65	IRQ 65	Level	
	rsp_ch4_s	98	SPI 66	IRQ 66	Level	
	rsp_ch5_s	99	SPI 67	IRQ 67	Level	
MHU (non-secure)	msg_ch0_ns	100	SPI 68	IRQ 68	Level	
	msg_ch1_ns	101	SPI 69	IRQ 69	Level	
	msg_ch2_ns	102	SPI 70	IRQ 70	Level	
	msg_ch3_ns	103	SPI 71	IRQ 71	Level	
	msg_ch4_ns	104	SPI 72	IRQ 72	Level	
	msg_ch5_ns	105	SPI 73	IRQ 73	Level	
	rsp_ch0_ns	106	SPI 74	IRQ 74	Level	
	rsp_ch1_ns	107	SPI 75	IRQ 75	Level	
	rsp_ch2_ns	108	SPI 76	IRQ 76	Level	
	rsp_ch3_ns	109	SPI 77	IRQ 77	Level	
	rsp_ch4_ns	110	SPI 78	IRQ 78	Level	
	rsp_ch5_ns	111	SPI 79	IRQ 79	Level	
MHU (software)	sw_mhu_int_0	112	SPI 80	IRQ 80	Level	
	sw_mhu_int_1	113	SPI 81	IRQ 81	Level	
	sw_mhu_int_2	114	SPI 82	IRQ 82	Level	
	sw_mhu_int_3	115	SPI 83	IRQ 83	Level	

Table 8.2 Interrupt mapping (4/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
GbE (ch0)	pif_int_n_ch0	116	SPI 84	IRQ 84	Level	
	int_fil_n_ch0	117	SPI 85	IRQ 85	Level	
	int_arp_ns_n_ch0	118	SPI 86	IRQ 86	Level	
GbE (ch1)	pif_int_n_ch1	119	SPI 87	IRQ 87	Level	
	int_fil_n_ch1	120	SPI 88	IRQ 88	Level	
	int_arp_ns_n_ch!	121	SPI 89	IRQ 89	Level	
USB2.0 (Host ch0)	U2H0_INT	122	SPI 90	IRQ 90	Level	
	U2H0_OHCI_INT	123	SPI 91	IRQ 91	Level	
	U2H0_EHCI_INT	124	SPI 92	IRQ 92	Level	
	U2H0_WAKEON_INT	125	SPI 93	IRQ 93	Level	
	U2H0_OBINT	126	SPI 94	IRQ 94	Level	
USB2.0 (Host ch1)	U2H1_INT	127	SPI 95	IRQ 95	Level	
	U2H1_OHCI_INT	128	SPI 96	IRQ 96	Level	
	U2H1_EHCI_INT	129	SPI 97	IRQ 97	Level	
	U2H1_WAKEON_INT	130	SPI 98	IRQ 98	Level	
	U2H1_OBINT	131	SPI 99	IRQ 99	Level	
USB2.0 (Function)	U2P_IXL_INT	132	SPI 100	IRQ 100	Edge	
	U2P_INT_DMA[0]	133	SPI 101	IRQ 101	Level	
	U2P_INT_DMA[1]	134	SPI 102	IRQ 102	Level	
	U2P_INT_DMAERR	135	SPI 103	IRQ 103	Level	
SDHI/eMMC (ch0)	OXMNIRQ0	136	SPI 104	IRQ 104	Level	
	OXASIOIRQ0	137	SPI 105	IRQ 105	Level	
SDHI/eMMC (ch1)	OXMNIRQ1	138	SPI 106	IRQ 106	Level	
	OXASIOIRQ1	139	SPI 107	IRQ 107	Level	
DMAC (Secure)	DMAINT0_S	140	SPI 108	IRQ 108	Edge	
	DMAINT1_S	141	SPI 109	IRQ 109	Edge	
	DMAINT2_S	142	SPI 110	IRQ 110	Edge	
	DMAINT3_S	143	SPI 111	IRQ 111	Edge	
	DMAINT4_S	144	SPI 112	IRQ 112	Edge	
	DMAINT5_S	145	SPI 113	IRQ 113	Edge	
	DMAINT6_S	146	SPI 114	IRQ 114	Edge	
	DMAINT7_S	147	SPI 115	IRQ 115	Edge	
	DMAINT8_S	148	SPI 116	IRQ 116	Edge	
	DMAINT9_S	149	SPI 117	IRQ 117	Edge	
	DMAINT10_S	150	SPI 118	IRQ 118	Edge	
	DMAINT11_S	151	SPI 119	IRQ 119	Edge	
	DMAINT12_S	152	SPI 120	IRQ 120	Edge	
	DMAINT13_S	153	SPI 121	IRQ 121	Edge	
	DMAINT14_S	154	SPI 122	IRQ 122	Edge	
	DMAINT15_S	155	SPI 123	IRQ 123	Edge	
	DMAERR_S	156	SPI 124	IRQ 124	Edge	

Table 8.2 Interrupt mapping (5/13)

		Cortex-A55		Cortex-M33		
Interrupt Source*3	Cause of Interrupt	Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1	Interrupt Type*2	Note
DMAC (Non-Secure)	DMAINT0_NS	157	SPI 125	IRQ 125	Edge	
	DMAINT1_NS	158	SPI 126	IRQ 126	Edge	
	DMAINT2_NS	159	SPI 127	IRQ 127	Edge	
	DMAINT3_NS	160	SPI 128	IRQ 128	Edge	
	DMAINT4_NS	161	SPI 129	IRQ 129	Edge	
	DMAINT5_NS	162	SPI 130	IRQ 130	Edge	
	DMAINT6_NS	163	SPI 131	IRQ 131	Edge	
	DMAINT7_NS	164	SPI 132	IRQ 132	Edge	
	DMAINT8_NS	165	SPI 133	IRQ 133	Edge	
	DMAINT9_NS	166	SPI 134	IRQ 134	Edge	
	DMAINT10_NS	167	SPI 135	IRQ 135	Edge	
	DMAINT11_NS	168	SPI 136	IRQ 136	Edge	
	DMAINT12_NS	169	SPI 137	IRQ 137	Edge	
	DMAINT13_NS	170	SPI 138	IRQ 138	Edge	
	DMAINT14_NS	171	SPI 139	IRQ 139	Edge	
	DMAINT15_NS	172	SPI 140	IRQ 140	Edge	
	DMAERR_NS	173	SPI 141	IRQ 141	Edge	
DSI	dsi_int_seq0	174	SPI 142	IRQ 142	Level	
	dsi_int_seq1	175	SPI 143	IRQ 143	Level	
	dsi_int_vin1	176	SPI 144	IRQ 144	Level	
	dsi_int_rcv	177	SPI 145	IRQ 145	Level	
	dsi_int_ferr	178	SPI 146	IRQ 146	Level	
	dsi_int_ppi	179	SPI 147	IRQ 147	Level	
	dsi_int_debug	180	SPI 148	IRQ 148	Level	
LCDC	VSPD_INT	181	SPI 149	IRQ 149	Level	
Reserved	—	182	SPI 150	IRQ 150	—	
Reserved	—	183	SPI 151	IRQ 151	—	
LCDC	DU_INT	184	SPI 152	IRQ 152	Level	
3DGE	IRQGPU	185	SPI 153	IRQ 153	Level	
	IRQJOB	186	SPI 154	IRQ 154	Level	
	IRQMMU	187	SPI 155	IRQ 155	Level	
	IRQEVENT	188	SPI 156	IRQ 156	Level	
ISU	ISU_INT_FRE	189	SPI 157	IRQ 157	Level	
	ISU_INT_DESE	190	SPI 158	IRQ 158	Level	
	ISU_INT_STOPE	191	SPI 159	IRQ 159	Level	
	ISU_INT_ERR	192	SPI 160	IRQ 160	Level	
VCPL4	VCP.vcpl4.vint	193	SPI 161	IRQ 161	Level	
	VCP.vcpl4.cint	194	SPI 162	IRQ 162	Level	
	VCP.vcpl4.vedcint	195	SPI 163	IRQ 163	Level	
	VCP.vcpl4.cedcint	196	SPI 164	IRQ 164	Level	
	VCP.fcpcs.edcint	197	SPI 165	IRQ 165	Level	

Table 8.2 Interrupt mapping (6/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
CRU	csi2_link_int	198	SPI 166	IRQ 166	Level	
	image_conv_int	199	SPI 167	IRQ 167	Level	
	image_conv_err_int	200	SPI 168	IRQ 168	Level	
	axi_mst_err_int	201	SPI 169	IRQ 169	Level	
MTU3a (ch0)	TGIA0	202	SPI 170	IRQ 170	Edge	
	TGIB0	203	SPI 171	IRQ 171	Edge	
	TGIC0	204	SPI 172	IRQ 172	Edge	
	TGID0	205	SPI 173	IRQ 173	Edge	
	TCIV0	206	SPI 174	IRQ 174	Edge	
	TGIE0	207	SPI 175	IRQ 175	Edge	
	TGIF0	208	SPI 176	IRQ 176	Edge	
MTU3a (ch1)	TGIA1	209	SPI 177	IRQ 177	Edge	
	TGIB1	210	SPI 178	IRQ 178	Edge	
	TCIV1	211	SPI 179	IRQ 179	Edge	
	TCIU1	212	SPI 180	IRQ 180	Edge	
MTU3a (ch2)	TGIA2	213	SPI 181	IRQ 181	Edge	
	TGIB2	214	SPI 182	IRQ 182	Edge	
	TCIV2	215	SPI 183	IRQ 183	Edge	
	TCIU2	216	SPI 184	IRQ 184	Edge	
MTU3a (ch3)	TGIA3	217	SPI 185	IRQ 185	Edge	
	TGIB3	218	SPI 186	IRQ 186	Edge	
	TGIC3	219	SPI 187	IRQ 187	Edge	
	TGID3	220	SPI 188	IRQ 188	Edge	
	TCIV3	221	SPI 189	IRQ 189	Edge	
MTU3a (ch4)	TGIA4	222	SPI 190	IRQ 190	Edge	
	TGIB4	223	SPI 191	IRQ 191	Edge	
	TGIC4	224	SPI 192	IRQ 192	Edge	
	TGID4	225	SPI 193	IRQ 193	Edge	
	TCIV4	226	SPI 194	IRQ 194	Edge	
MTU3a (ch5)	TGIU5	227	SPI 195	IRQ 195	Edge	
	TGIV5	228	SPI 196	IRQ 196	Edge	
	TGIW5	229	SPI 197	IRQ 197	Edge	
MTU3a (ch6)	TGIA6	230	SPI 198	IRQ 198	Edge	
	TGIB6	231	SPI 199	IRQ 199	Edge	
	TGIC6	232	SPI 200	IRQ 200	Edge	
	TGID6	233	SPI 201	IRQ 201	Edge	
	TCIV6	234	SPI 202	IRQ 202	Edge	
MTU3a (ch7)	TGIA7	235	SPI 203	IRQ 203	Edge	
	TGIB7	236	SPI 204	IRQ 204	Edge	
	TGIC7	237	SPI 205	IRQ 205	Edge	
	TGID7	238	SPI 206	IRQ 206	Edge	
	TCIV7	239	SPI 207	IRQ 207	Edge	

Table 8.2 Interrupt mapping (7/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
MTU3a (ch8)	TGIA8	240	SPI 208	IRQ 208	Edge	
	TGIB8	241	SPI 209	IRQ 209	Edge	
	TGIC8	242	SPI 210	IRQ 210	Edge	
	TGID8	243	SPI 211	IRQ 211	Edge	
	TCIV8	244	SPI 212	IRQ 212	Edge	
	TCIU8	245	SPI 213	IRQ 213	Edge	
POE3	OEI1	246	SPI 214	IRQ 214	Level	
	OEI2	247	SPI 215	IRQ 215	Level	
	OEI3	248	SPI 216	IRQ 216	Level	
	OEI4	249	SPI 217	IRQ 217	Level	
GPT (ch0)	CCMPA0	250	SPI 218	IRQ 218	Edge	
	CCMPB0	251	SPI 219	IRQ 219	Edge	
	CMPC0	252	SPI 220	IRQ 220	Edge	
	CMPD0	253	SPI 221	IRQ 221	Edge	
	CMPE0	254	SPI 222	IRQ 222	Edge	
	CMPF0	255	SPI 223	IRQ 223	Edge	
	ADTRGA0	256	SPI 224	IRQ 224	Edge	
	ADTRGB0	257	SPI 225	IRQ 225	Edge	
	OVF0	258	SPI 226	IRQ 226	Edge	
	UNF0	259	SPI 227	IRQ 227	Edge	
Reserved	—	260	SPI 228	IRQ 228	—	
Reserved	—	261	SPI 229	IRQ 229	—	
Reserved	—	262	SPI 230	IRQ 230	—	
GPT (ch1)	CCMPA1	263	SPI 231	IRQ 231	Edge	
	CCMPB1	264	SPI 232	IRQ 232	Edge	
	CMPC1	265	SPI 233	IRQ 233	Edge	
	CMPD1	266	SPI 234	IRQ 234	Edge	
	CMPE1	267	SPI 235	IRQ 235	Edge	
	CMPF1	268	SPI 236	IRQ 236	Edge	
	ADTRGA1	269	SPI 237	IRQ 237	Edge	
	ADTRGB1	270	SPI 238	IRQ 238	Edge	
	OVF1	271	SPI 239	IRQ 239	Edge	
	UNF1	272	SPI 240	IRQ 240	Edge	
Reserved	—	273	SPI 241	IRQ 241	—	
Reserved	—	274	SPI 242	IRQ 242	—	
Reserved	—	275	SPI 243	IRQ 243	—	

Table 8.2 Interrupt mapping (8/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
GPT (ch2)	CCMPA2	276	SPI 244	IRQ 244	Edge	
	CCMPB2	277	SPI 245	IRQ 245	Edge	
	CMPC2	278	SPI 246	IRQ 246	Edge	
	CMPD2	279	SPI 247	IRQ 247	Edge	
	CMPE2	280	SPI 248	IRQ 248	Edge	
	CMPF2	281	SPI 249	IRQ 249	Edge	
	ADTRGA2	282	SPI 250	IRQ 250	Edge	
	ADTRGB2	283	SPI 251	IRQ 251	Edge	
	OVF2	284	SPI 252	IRQ 252	Edge	
	UNF2	285	SPI 253	IRQ 253	Edge	
Reserved	—	286	SPI 254	IRQ 254	—	
Reserved	—	287	SPI 255	IRQ 255	—	
Reserved	—	288	SPI 256	IRQ 256	—	
GTP (ch3)	CCMPA3	289	SPI 257	IRQ 257	Edge	
	CCMPB3	290	SPI 258	IRQ 258	Edge	
	CMPC3	291	SPI 259	IRQ 259	Edge	
	CMPD3	292	SPI 260	IRQ 260	Edge	
	CMPE3	293	SPI 261	IRQ 261	Edge	
	CMPF3	294	SPI 262	IRQ 262	Edge	
	ADTRGA3	295	SPI 263	IRQ 263	Edge	
	ADTRGB3	296	SPI 264	IRQ 264	Edge	
	OVF3	297	SPI 265	IRQ 265	Edge	
	UNF3	298	SPI 266	IRQ 266	Edge	
Reserved	—	299	SPI 267	IRQ 267	—	
Reserved	—	300	SPI 268	IRQ 268	—	
Reserved	—	301	SPI 269	IRQ 269	—	
GPT (ch4)	CCMPA4	302	SPI 270	IRQ 270	Edge	
	CCMPB4	303	SPI 271	IRQ 271	Edge	
	CMPC4	304	SPI 272	IRQ 272	Edge	
	CMPD4	305	SPI 273	IRQ 273	Edge	
	CMPE4	306	SPI 274	IRQ 274	Edge	
	CMPF4	307	SPI 275	IRQ 275	Edge	
	ADTRGA4	308	SPI 276	IRQ 276	Edge	
	ADTRGB4	309	SPI 277	IRQ 277	Edge	
	OVF4	310	SPI 278	IRQ 278	Edge	
	UNF4	311	SPI 279	IRQ 279	Edge	
Reserved	—	312	SPI 280	IRQ 280	—	
Reserved	—	313	SPI 281	IRQ 281	—	
Reserved	—	314	SPI 282	IRQ 282	—	

Table 8.2 Interrupt mapping (9/13)

		Cortex-A55		Cortex-M33		
Interrupt Source*3	Cause of Interrupt	Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1	Interrupt Type*2	Note
GPT (ch5)	CCMPA5	315	SPI 283	IRQ 283	Edge	
	CCMPB5	316	SPI 284	IRQ 284	Edge	
	CMPC5	317	SPI 285	IRQ 285	Edge	
	CMPD5	318	SPI 286	IRQ 286	Edge	
	CMPE5	319	SPI 287	IRQ 287	Edge	
	CMPF5	320	SPI 288	IRQ 288	Edge	
	ADTRGA5	321	SPI 289	IRQ 289	Edge	
	ADTRGB5	322	SPI 290	IRQ 290	Edge	
	OVF5	323	SPI 291	IRQ 291	Edge	
	UNF5	324	SPI 292	IRQ 292	Edge	
Reserved	—	325	SPI 293	IRQ 293	—	
Reserved	—	326	SPI 294	IRQ 294	—	
Reserved	—	327	SPI 295	IRQ 295	—	
GPT (ch6)	CCMPA6	328	SPI 296	IRQ 296	Edge	
	CCMPB6	329	SPI 297	IRQ 297	Edge	
	CMPC6	330	SPI 298	IRQ 298	Edge	
	CMPD6	331	SPI 299	IRQ 299	Edge	
	CMPE6	332	SPI 300	IRQ 300	Edge	
	CMPF6	333	SPI 301	IRQ 301	Edge	
	ADTRGA6	334	SPI 302	IRQ 302	Edge	
	ADTRGB6	335	SPI 303	IRQ 303	Edge	
	OVF6	336	SPI 304	IRQ 304	Edge	
	UNF6	337	SPI 305	IRQ 305	Edge	
Reserved	—	338	SPI 306	IRQ 306	—	
Reserved	—	339	SPI 307	IRQ 307	—	
Reserved	—	340	SPI 308	IRQ 308	—	
GPT (ch7)	CCMPA7	341	SPI 309	IRQ 309	Edge	
	CCMPB7	342	SPI 310	IRQ 310	Edge	
	CMPC7	343	SPI 311	IRQ 311	Edge	
	CMPD7	344	SPI 312	IRQ 312	Edge	
	CMPE6	345	SPI 313	IRQ 313	Edge	
	CMPF7	346	SPI 314	IRQ 314	Edge	
	ADTRGA7	347	SPI 315	IRQ 315	Edge	
	ADTRGB7	348	SPI 316	IRQ 316	Edge	
	OVF6	349	SPI 317	IRQ 317	Edge	
	UNF7	350	SPI 318	IRQ 318	Edge	
Reserved	—	351	SPI 319	IRQ 319	—	
Reserved	—	352	SPI 320	IRQ 320	—	
Reserved	—	353	SPI 321	IRQ 321	—	
POEGA	GROUP0	354	SPI 322	IRQ 322	Level	
POEGB	GROUP1	355	SPI 323	IRQ 323	Level	
POEGC	GROUP2	356	SPI 324	IRQ 324	Level	
POEGD	GROUP3	357	SPI 325	IRQ 325	Level	

Table 8.2 Interrupt mapping (10/13)

		Cortex-A55		Cortex-M33		
Interrupt Source*3	Cause of Interrupt	Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1	Interrupt Type*2	Note
SSIF (ch0)	INT_ssif_int_req_0	358	SPI 326	IRQ 326	Level	
	INT_ssif_dma_rx_0	359	SPI 327	IRQ 327	Edge	
	INT_ssif_dma_tx_0	360	SPI 328	IRQ 328	Edge	
Reserved	—	361	SPI 329	IRQ 329	—	
SSIF (ch1)	INT_ssif_int_req_1	362	SPI 330	IRQ 330	Level	
	INT_ssif_dma_rx_1	363	SPI 331	IRQ 331	Edge	
	INT_ssif_dma_tx_1	364	SPI 332	IRQ 332	Edge	
Reserved	—	365	SPI 333	IRQ 333	—	
SSIF (ch2)	INT_ssif_int_req_2	366	SPI 334	IRQ 334	Level	
Reserved	—	367	SPI 335	IRQ 335	—	
Reserved	—	368	SPI 336	IRQ 336	—	
SSIF (ch2)	INT_ssif_dma_rt_2	369	SPI 337	IRQ 337	Edge	
SSIF (ch3)	INT_ssif_int_req_3	370	SPI 338	IRQ 338	Level	
	INT_ssif_dma_rx_3	371	SPI 339	IRQ 339	Edge	
	INT_ssif_dma_tx_3	372	SPI 340	IRQ 340	Edge	
Reserved	—	373	SPI 341	IRQ 341	—	
SRC	SRC_IDEI	374	SPI 342	IRQ 342	Edge	
	SRC_ODFI	375	SPI 343	IRQ 343	Edge	
	SRC_CEF	376	SPI 344	IRQ 344	Level	
	SRC_UDF	377	SPI 345	IRQ 345	Level	
	SRC_OVF	378	SPI 346	IRQ 346	Level	
ADC	INTAD	379	SPI 347	IRQ 347	Edge	
I2C (ch0)	INTRIICRI0	380	SPI 348	IRQ 348	Edge	
	INTRIICTI0	381	SPI 349	IRQ 349	Edge	
	INTRIICTEI0	382	SPI 350	IRQ 350	Level	
	INTRIICNAKI0	383	SPI 351	IRQ 351	Level	
	INTRIICSPI0	384	SPI 352	IRQ 352	Level	
	INTRIICSTI0	385	SPI 353	IRQ 353	Level	
	INTRIICALI0	386	SPI 354	IRQ 354	Level	
	INTRIICTMOI0	387	SPI 355	IRQ 355	Level	
I2C (ch1)	INTRIICRI1	388	SPI 356	IRQ 356	Edge	
	INTRIICTI1	389	SPI 357	IRQ 357	Edge	
	INTRIICTEI1	390	SPI 358	IRQ 358	Level	
	INTRIICNAKI1	391	SPI 359	IRQ 359	Level	
	INTRIICSPI1	392	SPI 360	IRQ 360	Level	
	INTRIICSTI1	393	SPI 361	IRQ 361	Level	
	INTRIICALI1	394	SPI 362	IRQ 362	Level	
	INTRIICTMOI1	395	SPI 363	IRQ 363	Level	

Table 8.2 Interrupt mapping (11/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
I2C (ch2)	INTRIICRI2	396	SPI 364	IRQ 364	Edge	
	INTRIICTI2	397	SPI 365	IRQ 365	Edge	
	INTRIICTEI2	398	SPI 366	IRQ 366	Level	
	INTRIICNAKI2	399	SPI 367	IRQ 367	Level	
	INTRIICSPI2	400	SPI 368	IRQ 368	Level	
	INTRIICSTI2	401	SPI 369	IRQ 369	Level	
	INTRIICALI2	402	SPI 370	IRQ 370	Level	
	INTRIICTMOI2	403	SPI 371	IRQ 371	Level	
I2C (ch3)	INTRIICRI3	404	SPI 372	IRQ 372	Edge	
	INTRIICTI3	405	SPI 373	IRQ 373	Edge	
	INTRIICTEI3	406	SPI 374	IRQ 374	Level	
	INTRIICNAKI3	407	SPI 375	IRQ 375	Level	
	INTRIICSPI3	408	SPI 376	IRQ 376	Level	
	INTRIICSTI3	409	SPI 377	IRQ 377	Level	
	INTRIICALI3	410	SPI 378	IRQ 378	Level	
	INTRIICTMOI3	411	SPI 379	IRQ 379	Level	
SCIFA (ch0)	ERI0	412	SPI 380	IRQ 380	Level	
	BRI0	413	SPI 381	IRQ 381	Level	
	RXI0	414	SPI 382	IRQ 382	Level	
	TXI0	415	SPI 383	IRQ 383	Level	
	TEI0_DRI0	416	SPI 384	IRQ 384	Level	
SCIFA (ch1)	ERI1	417	SPI 385	IRQ 385	Level	
	BRI1	418	SPI 386	IRQ 386	Level	
	RXI1	419	SPI 387	IRQ 387	Level	
	TXI1	420	SPI 388	IRQ 388	Level	
	TEI1_DRI1	421	SPI 389	IRQ 389	Level	
SCIFA (ch2)	ERI2	422	SPI 390	IRQ 390	Level	
	BRI2	423	SPI 391	IRQ 391	Level	
	RXI2	424	SPI 392	IRQ 392	Level	
	TXI2	425	SPI 393	IRQ 393	Level	
	TEI2_DRI2	426	SPI 394	IRQ 394	Level	
SCIFA (ch3)	ERI3	427	SPI 395	IRQ 395	Level	
	BRI3	428	SPI 396	IRQ 396	Level	
	RXI3	429	SPI 397	IRQ 397	Level	
	TXI3	430	SPI 398	IRQ 398	Level	
	TEI3_DRI3	431	SPI 399	IRQ 399	Level	
SCIFA (ch4)	ERI4	432	SPI 400	IRQ 400	Level	
	BRI4	433	SPI 401	IRQ 401	Level	
	RXI4	434	SPI 402	IRQ 402	Level	
	TXI4	435	SPI 403	IRQ 403	Level	
	TEI4_DRI4	436	SPI 404	IRQ 404	Level	

Table 8.2 Interrupt mapping (12/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
SCIg (ch0)	ERI0	437	SPI 405	IRQ 405	Level	
	RXI0	438	SPI 406	IRQ 406	Edge	
	TXI0	439	SPI 407	IRQ 407	Edge	
	TEI0	440	SPI 408	IRQ 408	Level	
SCIg (ch1)	ERI1	441	SPI 409	IRQ 409	Level	
	RXI1	442	SPI 410	IRQ 410	Edge	
	TXI1	443	SPI 411	IRQ 411	Edge	
	TEI1	444	SPI 412	IRQ 412	Level	
RSPI (ch0)	SPRI0	445	SPI 413	IRQ 413	Level	
	SPTI0	446	SPI 414	IRQ 414	Level	
	SPEI0	447	SPI 415	IRQ 415	Level	
RSPI (ch1)	SPRI1	448	SPI 416	IRQ 416	Level	
	SPTI1	449	SPI 417	IRQ 417	Level	
	SPEI1	450	SPI 418	IRQ 418	Level	
RSPI (ch2)	SPRI2	451	SPI 419	IRQ 419	Level	
	SPTI2	452	SPI 420	IRQ 420	Level	
	SPEI2	453	SPI 421	IRQ 421	Level	
CANFD	INTRCAN0ERR	454	SPI 422	IRQ 422	Level	
	INTRCAN1ERR	455	SPI 423	IRQ 423	Level	
	INTRCAN0REC	456	SPI 424	IRQ 424	Level	
	INTRCAN1REC	457	SPI 425	IRQ 425	Level	
	INTRCANGERR	458	SPI 426	IRQ 426	Level	
	INTRCANGRECC	459	SPI 427	IRQ 427	Level	
	INTRCAN0TRX	460	SPI 428	IRQ 428	Level	
	INTRCAN1TRX	461	SPI 429	IRQ 429	Level	
TSIP	WRRDY1	462	SPI 430	IRQ 430	Edge	
	WRRDY0	463	SPI 431	IRQ 431	Edge	
	WRRDY4	464	SPI 432	IRQ 432	Edge	
	RDRDY1	465	SPI 433	IRQ 433	Edge	
	RDRDY0	466	SPI 434	IRQ 434	Edge	
	ROMOK	467	SPI 435	IRQ 435	Edge	
	LONG_PLG	468	SPI 436	IRQ 436	Edge	
	PROC_BUSY	469	SPI 437	IRQ 437	Edge	
	IRDRDY	470	SPI 438	IRQ 438	Edge	
	IWRRDY	471	SPI 439	IRQ 439	Edge	
Reserved	—	472	SPI 440	IRQ 440	—	
Reserved	—	473	SPI 441	IRQ 441	—	
Reserved	—	474	SPI 442	IRQ 442	—	
Reserved	—	475	SPI 443	IRQ 443	—	

Table 8.2 Interrupt mapping (13/13)

Interrupt Source*3	Cause of Interrupt	Cortex-A55		Cortex-M33	Interrupt Type*2	Note
		Interrupt ID	SGI,PPI,SPI No.	IRQ No.*1		
GPIO Interrupt	TINT0	476	SPI 444	IRQ 444	Level	*6
	TINT1	477	SPI 445	IRQ 445	Level	*6
	TINT2	478	SPI 446	IRQ 446	Level	*6
	TINT3	479	SPI 447	IRQ 447	Level	*6
	TINT4	480	SPI 448	IRQ 448	Level	*6
	TINT5	481	SPI 449	IRQ 449	Level	*6
	TINT6	482	SPI 450	IRQ 450	Level	*6
	TINT7	483	SPI 451	IRQ 451	Level	*6
	TINT8	484	SPI 452	IRQ 452	Level	*6
	TINT9	485	SPI 453	IRQ 453	Level	*6
	TINT10	486	SPI 454	IRQ 454	Level	*6
	TINT11	487	SPI 455	IRQ 455	Level	*6
	TINT12	488	SPI 456	IRQ 456	Level	*6
	TINT13	489	SPI 457	IRQ 457	Level	*6
	TINT14	490	SPI 458	IRQ 458	Level	*6
	TINT15	491	SPI 459	IRQ 459	Level	*6
	TINT16	492	SPI 460	IRQ 460	Level	*6
	TINT17	493	SPI 461	IRQ 461	Level	*6
	TINT18	494	SPI 462	IRQ 462	Level	*6
	TINT19	495	SPI 463	IRQ 463	Level	*6
	TINT20	496	SPI 464	IRQ 464	Level	*6
	TINT21	497	SPI 465	IRQ 465	Level	*6
	TINT22	498	SPI 466	IRQ 466	Level	*6
	TINT23	499	SPI 467	IRQ 467	Level	*6
	TINT24	500	SPI 468	IRQ 468	Level	*6
	TINT25	501	SPI 469	IRQ 469	Level	*6
	TINT26	502	SPI 470	IRQ 470	Level	*6
	TINT27	503	SPI 471	IRQ 471	Level	*6
	TINT28	504	SPI 472	IRQ 472	Level	*6
	TINT29	505	SPI 473	IRQ 473	Level	*6
	TINT30	506	SPI 474	IRQ 474	Level	*6
	TINT31	507	SPI 475	IRQ 475	Level	*6
Reserved	—	508	SPI 476	IRQ 476	—	
Reserved	—	509	SPI 477	IRQ 477	—	
Reserved	—	510	SPI 478	IRQ 478	—	
Reserved	—	511	SPI 479	IRQ 479	—	

Note 1. Exception handlers of Cortex-M33 has Interrupt Service Routines (ISRs), Fault handler and System handler.
IRQ0-479 is treated as Interrupt Service Routines (ISRs).
Refer to “Arm Cortex-M33 Processor Technical Reference Manual” and “Arm Cortex-M33 Processor User Guide” for details and targets of Fault handler and System handler.

Note 2. Set GICD_ICFGR<n> of the interrupt controller for Cortex-A55 (GIC-600) according to the description of Interrupt Type for each Interrupt ID.
Refer to “ARM CoreLink GIC-600 Generic Interrupt Controller Technical Reference Manual” for the specification of GIC-600 registers.
Equivalent settings are not required for the interrupt controller for Cortex-M33 (Cortex-M33 built-in NVIC).

Note 3. Refer to the source unit section for the specification of each interrupt source.

Note 4. NMI is not treated as NMI exception.
NMI interrupt setting can be controlled by the registers in IA55 and IM33. Refer to **Section 8.7.1** for details.

Note 5. IRQ0-7 interrupt settings can be controlled by the registers in IA55 and IM33. Refer to **Section 8.7.2** for details.

Note 6. TINT0-31 interrupt settings can be controlled by the registers in IA55 and IM33. Refer to **Section 8.7.3** for details.

Note 7. BUS_ERR_INT is the interrupt integrated multiple interrupts into one in IA55 and IM33.
The interrupt settings can be controlled by the registers in IA55 and IM33. Refer to **Section 8.7.4.1** for details.

Note 8. Each EC7TIE1, TC7TIE2 and EC7TIOVF is the interrupt integrated multiple interrupts into one in IA55 and IM33.
The interrupt settings can be controlled by the registers in IA55 and IM33. Refer to **Section 8.7.4.2** for details.

8.3 GIC-600 and NVIC Register Configuration

■ GIC-600 Register

Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual.
ITS and LPI are not supported.

Base Address: H’0_1190_0000 (Cortex-A55 Address Space)

Base Address: H’4190_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’5190_0000 (Cortex-M33 Address Space Secure)

■ NVIC Register

Refer to Arm® Cortex®-M33 Processor Revision: r0p4 Technical Reference Manual.

8.4 GIC-600 and NVIC Register Descriptions

■ GIC-600 Register

Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual for detail.

■ NVIC Register

Refer to Arm® Cortex®-M33 Processor Revision: r0p4 Technical Reference Manual for detail.

8.5 IA55 and IM33 Register Configuration

■ IA55

Base Address: H’0_110A_0000 (Cortex-A55 Address Space)

Base Address: H’410A_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’510A_0000 (Cortex-M33 Address Space Secure)

■ IM33

Base Address: H’0_110B_0000 (Cortex-A55 Address Space)

Base Address: H’410B_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’510B_0000 (Cortex-M33 Address Space Secure)

The following table shows IA55 and IM33 Register list.
Prohibit to write undefined area.

Table 8.3 IA55 and IM33 Register List

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
NMI Status Control Register	NSCR	R/W	H’0000_0000	H’0000	32
NMI Interrupt Type Selection Register	NITSR	R/W	H’0000_0000	H’0004	32
IRQ Status Control Register	ISCR	R/W	H’0000_0000	H’0010	32
IRQ Interrupt Type Selection Register	IITSR	R/W	H’0000_0000	H’0014	32
TINT Status Control Register	TSCR	R/W	H’0000_0000	H’0020	32
TINT Interrupt Type Selection Register0	TITSR0	R/W	H’0000_0000	H’0024	32
TINT Interrupt Type Selection Register1	TITSR1	R/W	H’0000_0000	H’0028	32
TINT Source Selection Register0	TSSR0	R/W	H’0000_0000	H’0030	32
TINT Source Selection Register1	TSSR1	R/W	H’0000_0000	H’0034	32
TINT Source Selection Register2	TSSR2	R/W	H’0000_0000	H’0038	32
TINT Source Selection Register3	TSSR3	R/W	H’0000_0000	H’003C	32
TINT Source Selection Register4	TSSR4	R/W	H’0000_0000	H’0040	32
TINT Source Selection Register5	TSSR5	R/W	H’0000_0000	H’0044	32
TINT Source Selection Register6	TSSR6	R/W	H’0000_0000	H’0048	32
TINT Source Selection Register7	TSSR7	R/W	H’0000_0000	H’004C	32
Bus Error Interrupt Status Control Register0	BEISR0	R/W	H’0000_0000	H’0050	32
Bus Error Interrupt Status Control Register1	BEISR1	R/W	H’0000_0000	H’0054	32
ECCRAM Error Interrupt Status Control Register0	EREISR0	R/W	H’0000_0000	H’0060	32
ECCRAM Error Interrupt Status Control Register1	EREISR1	R/W	H’0000_0000	H’0064	32

8.6 IA55 and IM33 Register Descriptions

8.6.1 NMI Status Control Register (NSCR)

This register shows NMI status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NSMON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NSTAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	NSMON	0	R	This bit indicates NMI pin signal level. [Read operation] 0: NMI pin is Low-level. 1: NMI pin is High-level. [Write operation] Invalid to write.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	NSTAT	0	R/W	This bit indicates NMI interrupt status. NMI interrupt status can be cleared writing 0 in case NSTAT is 1. [Read operation] 0: NMI interrupt is not detected. 1: NMI interrupt is detected. [Write operation] ● In case NSTAT is 1 0: NMI interrupt status is cleared. 1: Invalid to write. ● In case NSTAT is 0 Invalid to write.

8.6.2 NMI Interrupt Type Selection Register (NITSR)

This register selects NMI detecting method.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NTSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	NTSEL	0	R/W	This bit is used to select NMI interrupt detection method. 0: Falling-edge detection. 1: Rising-edge detection.

8.6.3 IRQ Status Control Register (ISCR)

This register shows IRQ status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ISTAT7	ISTAT6	ISTAT5	ISTAT4	ISTAT3	ISTAT2	ISTAT1	ISTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n	ISTATn	0	R/W	This bit indicates IRQn interrupt status. When the interrupt detection method is set to an edge type detection in IITSR, IRQn interrupt status can be cleared writing 0 in case ISTATn is 1. [Read operation] 0: IRQn interrupt is not detected. 1: IRQn interrupt is detected. [Write operation] When “Falling-edge detection”, “Rising-edge detection” or “Falling/Rising-edge detection” is set in ITSR.: ● In case ISTAT is 1 0: IRQn interrupt detection status is cleared. 1: Invalid to write. ● In case ISTAT is 0 Invalid to write. When “Low-level detection” is set in IITSR.: Invalid to write.

Note: n = 7 to 0

8.6.4 IRQ Interrupt Type Selection Register (IITSR)

This register selects IRQ detecting method.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IITSEL7		IITSEL6		IITSEL5		IITSEL4		IITSEL3		IITSEL2		IITSEL1		IITSEL0	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2n+1 to 2n	IITSEL	All 0	R/W	This bit is used to select IRQn interrupt detection method. 00: Low-level detection. 01: Falling-edge detection. 10: Rising-edge detection. 11: Falling/Rising-edge detection.

Note: n = 7 to 0

8.6.5 TINT Status Control Register (TSCR)

This register shows TINT status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSTAT31	TSTAT30	TSTAT29	TSTAT28	TSTAT27	TSTAT26	TSTAT25	TSTAT24	TSTAT23	TSTAT22	TSTAT21	TSTAT20	TSTAT19	TSTAT18	TSTAT17	TSTAT16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSTAT15	TSTAT14	TSTAT13	TSTAT12	TSTAT11	TSTAT10	TSTAT9	TSTAT8	TSTAT7	TSTAT6	TSTAT5	TSTAT4	TSTAT3	TSTAT2	TSTAT1	TSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
n	TSTATn	All 0	R/W	This bit indicates TINTn interrupt status. When the interrupt detection method is set to edge type detection in TITSR0/1, TINTn interrupt status can be cleared writing 0 in case TSTATn is 1. [Read operation] 0: TINTn interrupt is not detected. 1: TINTn interrupt is detected. [Write operation] When “Rising-edge detection” or “Falling-edge detection” is set in TITSR0/1.: ● In case TSTAT is 1 0: TINTn interrupt detection status is cleared. 1: Invalid to write. ● In case TSTAT is 0 Invalid to write. When “Low-level detection” or ”High-Level detection” is set in TITSR0/1.: Invalid to write.

Note: n = 31 to 0

8.6.6 TINT Interrupt Type Selection Register0 (TITSR0)

This register selects detecting method of TINT15 to TINT0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TITSEL15		TITSEL14		TITSEL13		TITSEL12		TITSEL11		TITSEL10		TITSEL9		TITSEL8	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TITSEL7		TITSEL6		TITSEL5		TITSEL4		TITSEL3		TITSEL2		TITSEL1		TITSEL0	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
2n+1 to 2n	TITSELn	All 0	R/W	This bit is used to select TINTn interrupt detection method. 00: Rising-edge detection. 01: Falling-edge detection. 10: High-level detection. 11: Low-level detection.

Note: n = 15 to 0

8.6.7 TINT Interrupt Type Selection Register1 (TITSR1)

This register selects detecting method of TINT31 to TINT16.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TITSEL31		TITSEL30		TITSEL29		TITSEL28		TITSEL27		TITSEL26		TITSEL25		TITSEL24	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TITSEL23		TITSEL22		TITSEL21		TITSEL20		TITSEL19		TITSEL18		TITSEL17		TITSEL16	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
2n+1 to 2n	TITSEL n+16	All 0	R/W	This bit is used to select TINTn+16 interrupt detection method. 00: Rising-edge detection. 01: Falling-edge detection. 10: High-level detection. 11: Low-level detection.

Note: n = 15 to 0

8.6.8 TINT Source Selection Register0 (TSSR0)

This register selects the interrupt source of TINT3 to TINT0.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN3	TSSEL3							TIEN2	TSSEL2						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN1	TSSEL1							TIEN0	TSSEL0						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8n+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8n+6 to 8n	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 3 to 0

8.6.9 TINT Source Selection Register1 (TSSR1)

This register selects the interrupt source of TINT7 to TINT4.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN7	TSSEL7							TIEN6	TSSEL6						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN5	TSSEL5							TIEN4	TSSEL4						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-4)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-4)+6 to 8(n-4)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 7 to 4

8.6.10 TINT Source Selection Register2 (TSSR2)

This register selects the interrupt source of TINT11 to TINT8.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN11	TSSEL11							TIEN10	TSSEL10						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN9	TSSEL9							TIEN8	TSSEL8						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-8)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-8)+6 to 8(n-8)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 11 to 8

8.6.11 TINT Source Selection Register3 (TSSR3)

This register selects the interrupt source of TINT15 to TINT12.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN15	TSSEL15							TIEN14	TSSEL14						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN13	TSSEL13							TIEN12	TSSEL12						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-12)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-12)+6 to 8(n-12)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 15 to 12

8.6.12 TINT Source Selection Register4 (TSSR4)

This register selects the interrupt source of TINT19 to TINT16.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN19	TSSEL19							TIEN18	TSSEL18						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN17	TSSEL17							TIEN16	TSSEL16						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-16)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-16)+6 to 8(n-16)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 19 to 16

8.6.13 TINT Source Selection Register5 (TSSR5)

This register selects the interrupt source of TINT23 to TINT20.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN23	TSSEL23							TIEN22	TSSEL22						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN21	TSSEL21							TIEN20	TSSEL20						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-20)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-20)+6 to 8(n-20)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 23 to 20

8.6.14 TINT Source Selection Register6 (TSSR6)

This register selects the interrupt source of TINT27 to TINT24.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN27	TSSEL27							TIEN26	TSSEL26						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN25	TSSEL25							TIEN24	TSSEL24						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-24)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-24)+6 to 8(n-24)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 27 to 24

8.6.15 TINT Source Selection Register7 (TSSR7)

This register selects the interrupt source of TINT31 to TINT28.

Refer to **Section 8.7.3** about mapping GPIOINT onto GPIO pin.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIEN31	TSSEL31							TIEN30	TSSEL30						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIEN29	TSSEL29							TIEN28	TSSEL28						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
8(n-28)+7	TIENn	0	R/W	This bit enables TINTn. 1: Enable. 0: Invalid.
8(n-28)+6 to 8(n-28)	TSSELn	All 0	R/W	This bit selects a source of TINTn. 0000000: TINTn = GPIOINT0 0000001: TINTn = GPIOINT1 ~ 1111010: TINTn = GPIOINT122 1111011: Cannot be set. 1111100: Cannot be set. 1111101: Cannot be set. 1111110: Cannot be set. 1111111: Cannot be set. Assign different sources to TINT0-31.

Note: n = 31 to 28

8.6.16 Bus Error Interrupt Status Control Register0 (BEISR0)

This register shows Bus error status.

Refer to LSI Internal Bus section and **Section 8.7.4.1** for bus error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BESTAT31	BESTAT30	BESTAT29	BESTAT28	BESTAT27	BESTAT26	BESTAT25	BESTAT24	BESTAT23	BESTAT22	BESTAT21	BESTAT20	BESTAT19	BESTAT18	BESTAT17	BESTAT16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BESTAT15	BESTAT14	BESTAT13	TSTAT12	BESTAT11	BESTAT10	BESTAT9	BESTAT8	BESTAT7	BESTAT6	BESTAT5	BESTAT4	BESTAT3	BESTAT2	BESTAT1	BESTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
n	BESTATn	All 0	R/W	This bit indicates BUSERR_INTn interrupt status. BUSERR_INTn interrupt status can be cleared writing 0 in case BESTAT is 1. [Read operation] 0: BUSERR_INTn is not detected. 1: BUSERR_INTn is detected. [Write operation] ● In case BESTATn is 1 0: BUSERR_INTn interrupt status is cleared. 1: Invalid to write. ● In case BESTATn is 0 Invalid to write.

Note: n = 31 to 0

8.6.17 Bus Error Interrupt Status Control Register1 (BEISR1)

This register shows Bus error status.

Refer to LSI Internal Bus section and **Section 8.7.4.1** for bus error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BESTAT44	BESTAT43	BESTAT42	BESTAT41	BESTAT40	BESTAT39	BESTAT38	BESTAT37	BESTAT36	BESTAT35	BESTAT34	BESTAT33	BESTAT32
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
n-32	BESTATn	All 0	R/W	This bit indicates BUSERR_INTn interrupt status. BUSERR_INTn interrupt status can be cleared writing 0 in case BESTAT is 1. [Read operation] 0: BUSERR_INTn is not detected. 1: BUSERR_INTn is detected. [Write operation] ● In case BESTATn is 1 0: BUSERR_INTn interrupt status is cleared. 1: Invalid to write. ● In case BESTATn is 0 Invalid to write.

Note: n = 44 to 32

8.6.18 ECCRAM Error Interrupt Status Control Register0 (EREISR0)

This register shows ECC Error status of ECCRAM0.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	OFSTAT7	OFSTAT6	OFSTAT5	OFSTAT4	OFSTAT3	OFSTAT2	OFSTAT1	OFSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	E2STAT7	E2STAT6	E2STAT5	E2STAT4	E2STAT3	E2STAT2	E2STAT1	E2STAT0	E1STAT7	E1STAT6	E1STAT5	E1STAT4	E1STAT3	E1STAT2	E1STAT1	E1STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	OFSTATn	All 0	R/W	This bit indicates ECCRAM0_OFn interrupt status. ECCRAM0_OFn interrupt status can be cleared writing 0 in case OFSTATn is 1. [Read operation] 0: ECCRAM0_OFn is not detected. 1: ECCRAM0_OFn is detected. [Write operation] ● In case OFSTATn is 1 0: ECCRAM0_OFn interrupt status is cleared. 1: Invalid to write. ● In case OFSTATn is 0 Invalid to write.
15 to 8	E2STATn	All 0	R/W	This bit indicates ECCRAM0_2En interrupt status. ECCRAM0_2En interrupt status can be cleared writing 0 in case E2STATn is 1. [Read operation] 0: ECCRAM0_2En is not detected. 1: ECCRAM0_2En is detected. Write operation: ● In case E2STATn is 1 0: ECCRAM0_2En interrupt status is cleared. 1: Invalid to write. ● In case E2STATn is 0 Invalid to write.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	E1STATn	All 0	R/W	This bit indicates ECCRAM0_1En interrupt status. ECCRAM0_1En interrupt status can be cleared writing 0 in case E1STATn is 1. [Read operation] 0: ECCRAM0_1En is not detected. 1: ECCRAM0_1En is detected. [Write operation] ● In case E1STATn is 1 0: ECCRAM0_1En interrupt status is cleared. 1: Invalid to write. ● In case E1STATn is 0 Invalid to write.

Note: n = 7 to 0

8.6.19 ECCRAM Error Interrupt Status Control Register1 (EREISR1)

This register shows ECC Error status of ECCRAM1.

Refer to **Section 8.7.4.2** for ECCRAM error interrupt cause.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	OFSTAT7	OFSTAT6	OFSTAT5	OFSTAT4	OFSTAT3	OFSTAT2	OFSTAT1	OFSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	E2STAT7	E2STAT6	E2STAT5	E2STAT4	E2STAT3	E2STAT2	E2STAT1	E2STAT0	E1STAT7	E1STAT6	E1STAT5	E1STAT4	E1STAT3	E1STAT2	E1STAT1	E1STAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	OFSTATn	All 0	R/W	This bit indicates ECCRAM1_OFn interrupt status. ECCRAM1_OFn interrupt status can be cleared writing 0 in case OFSTATn is 1. [Read operation] 0: ECCRAM1_OFn is not detected. 1: ECCRAM1_OFn is detected. [Write operation] ● In case OFSTATn is 1 0: ECCRAM1_OFn interrupt status is cleared. 1: Invalid to write. ● In case OFSTATn is 0 Invalid to write.
15 to 8	E2STATn	All 0	R/W	This bit indicates ECCRAM1_2En interrupt status. ECCRAM1_2En interrupt status can be cleared writing 0 in case E2STATn is 1. [Read operation] 0: ECCRAM1_2En is not detected. 1: ECCRAM1_2En is detected. [Write operation] ● In case E2STATn is 1 0: ECCRAM1_2En interrupt status is cleared. 1: Invalid to write. ● In case E2STATn is 0 Invalid to write.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	E1STATn	All 0	R/W	This bit indicates ECCRAM1_1En interrupt status. ECCRAM1_1En interrupt status can be cleared writing 0 in case E1STATn is 1. [Read operation] 0: ECCRAM1_1En is not detected. 1: ECCRAM1_1En is detected. [Write operation] ● In case E1STATn is 1 0: ECCRAM1_1En interrupt status is cleared. 1: Invalid to write. ● In case E1STATn is 0 Invalid to write.

Note: n = 7 to 0

8.7 Operation

8.7.1 NMI Interrupt

NMI interrupt is the highest priority interrupt by default even though it can be masked.

NMI is not treated as NMI exception.

[Setting]

- A noise filter function of NMI pin can be enabled by GPIO register setting.
  Refer to **Section 41.4.1, Operation for GPIO Function** for detail.
- NMI pin input is detected by edge. A detection method is set by NTSEL bit of NMI Interrupt Type Selection Register (NITSR) in IA55/IM33. It can be selected “Rising-edge detection” or “Falling-edge detection”.

[Status Control]

- NMI interrupt request can be confirmed by reading NSTAT bit of NMI Status Control Register (NSCR) in IA55/IM33.
  The interrupt request can be cleared by writing 0 to NSTAT bit.
- A signal level (High or Low) of NMI pin can be confirmed by reading NSMON bit of NMI Status Control Register (NSCR) in IA55/IM33.

8.7.2 IRQ Interrupt

IRQ interrupt is the interrupt from IRQ0-7 input pins.

[Setting]

- Using IRQ pins as interrupt must be mapped GPIO pins onto IRQ pins by GPIO register setting.
  Refer to **Section 41.4.2, Operation for Peripheral Function** for detail.
- A noise filter function of IRQ pins can be enabled by GPIO register setting.
  Refer to **Section 41.4.1, Operation for GPIO Function** for detail.
- IRQ pin inputs are detected by a level or an edge. A detection method is set by ITSEL bit of IRQ Interrupt Type Selection Register (IITSR) in IA55/IM33. It can be selected “Low-level detection”, “Falling-edge detection”, “Rising-edge detection” or “Falling/Rising-edge detection”.

[Status Control]

- IRQ interrupt requests can be confirmed by reading ISTATn bit of IRQ Status Control Register (ISCR) in IA55/IM33.
- In case selected a level detection in IITSR, the interrupt requests are cleared when the interrupt sources de-assert the interrupts.
- In case selected an edge detection in IITSR, the interrupt requests can be cleared by writing 0 to ISTATn bit.

8.7.3 GPIO Interrupt (TINT)

GPIO interrupt is the interrupt using GPIO pins as external interrupt input pins.

[Setting]

- When using GPIO pins as interrupts, assign GPIO pins as external interrupt input pins (GPIOINT0-122) by GPIO register setting.
  Refer to **Section 41.4.1, Operation for GPIO Function** for details.
  Refer to **Table 8.4** for mapping GPIO pins onto GPIOINT0-122.
- The noise filter function of GPIO pins assigned to TINT can be enabled by GPIO register setting.
  Refer to **Section 41.4.1, Operation for GPIO Function** for details.
- Assign any 32 interrupt sources to TINT from GPIOINT0-122 by TINT Source Selection Register (TSSR0-7) in IA55/IM33.

  Note: Even if the TINT is assigned by TSSR0-7, no interrupt request is generated unless it is assigned to an external pin interrupt by GPIO registers.
- TINT inputs are detected by a level or an edge. A detection method is set by TTSEL bit of TINT Interrupt Type Selection Register (TITSR0/1) in IA55/IM33. It can be selected “Rising-edge detection”, “Falling-edge detection”, “High-level detection” or “Low-level detection”.

[Status Control]

- TINT interrupt requests can be confirmed by reading TSTATn bit of TINT Status Control Register (TSCR) in IA55/IM33.
- In case selected a level detection in TITSR0/1, the interrupt requests are cleared when the interrupt sources de-assert the interrupts.
- In case selected an edge detection in TITSR0/1, the interrupt requests can be cleared by writing 0 to TSTATn bit.

Table 8.4 GPIOINT Mapping

No.	GPIO Pin	GPIOINT
1	P0_0	GPIOINT0
2	P0_1	GPIOINT1
3	P1_0	GPIOINT2
4	P1_1	GPIOINT3
5	P2_0	GPIOINT4
6	P2_1	GPIOINT5
7	P3_0	GPIOINT6
8	P3_1	GPIOINT7
9	P4_0	GPIOINT8
10	P4_1	GPIOINT9
11	P5_0	GPIOINT10
12	P5_1	GPIOINT11
13	P5_2	GPIOINT12
14	P6_0	GPIOINT13
15	P6_1	GPIOINT14
16	P7_0	GPIOINT15
17	P7_1	GPIOINT16
18	P7_2	GPIOINT17
19	P8_0	GPIOINT18
20	P8_1	GPIOINT19
21	P8_2	GPIOINT20
22	P9_0	GPIOINT21
23	P9_1	GPIOINT22
24	P10_0	GPIOINT23
25	P10_1	GPIOINT24
26	P11_0	GPIOINT25
27	P11_1	GPIOINT26
28	P12_0	GPIOINT27
29	P12_1	GPIOINT28
30	P13_0	GPIOINT29
31	P13_1	GPIOINT30
32	P13_2	GPIOINT31
33	P14_0	GPIOINT32
34	P14_1	GPIOINT33
35	P15_0	GPIOINT34
36	P15_1	GPIOINT35
37	P16_0	GPIOINT36
38	P16_1	GPIOINT37
39	P17_0	GPIOINT38
40	P17_1	GPIOINT39
41	P17_2	GPIOINT40
42	P18_0	GPIOINT41
43	P18_1	GPIOINT42
44	P19_0	GPIOINT43
45	P19_1	GPIOINT44
46	P20_0	GPIOINT45
47	P20_1	GPIOINT46
48	P20_2	GPIOINT47
49	P21_0	GPIOINT48
50	P21_1	GPIOINT49
51	P22_0	GPIOINT50
52	P22_1	GPIOINT51
53	P23_0	GPIOINT52
54	P23_1	GPIOINT53
55	P24_0	GPIOINT54
56	P24_1	GPIOINT55
57	P25_0	GPIOINT56
58	P25_1	GPIOINT57
59	P26_0	GPIOINT58
60	P26_1	GPIOINT59
61	P27_0	GPIOINT60
62	P27_1	GPIOINT61
63	P28_0	GPIOINT62
64	P28_1	GPIOINT63
65	P29_0	GPIOINT64
66	P29_1	GPIOINT65
67	P30_0	GPIOINT66
68	P30_1	GPIOINT67
69	P31_0	GPIOINT68
70	P31_1	GPIOINT69
71	P32_0	GPIOINT70
72	P32_1	GPIOINT71
73	P33_0	GPIOINT72
74	P33_1	GPIOINT73
75	P34_0	GPIOINT74
76	P34_1	GPIOINT75
77	P35_0	GPIOINT76
78	P35_1	GPIOINT77
79	P36_0	GPIOINT78
80	P36_1	GPIOINT79
81	P37_0	GPIOINT80
82	P37_1	GPIOINT81
83	P37_2	GPIOINT82
84	P38_0	GPIOINT83
85	P38_1	GPIOINT84
86	P39_0	GPIOINT85
87	P39_1	GPIOINT86
88	P39_2	GPIOINT87
89	P40_0	GPIOINT88
90	P40_1	GPIOINT89
91	P40_2	GPIOINT90
92	P41_0	GPIOINT91
93	P41_1	GPIOINT92
94	P42_0	GPIOINT93
95	P42_1	GPIOINT94
96	P42_2	GPIOINT95
97	P42_3	GPIOINT96
98	P42_4	GPIOINT97
99	P43_0	GPIOINT98
100	P43_1	GPIOINT99
101	P43_2	GPIOINT100
102	P43_3	GPIOINT101
103	P44_0	GPIOINT102
104	P44_1	GPIOINT103
105	P44_2	GPIOINT104
106	P44_3	GPIOINT105
107	P45_0	GPIOINT106
108	P45_1	GPIOINT107
109	P45_2	GPIOINT108
110	P45_3	GPIOINT109
111	P46_0	GPIOINT110
112	P46_1	GPIOINT111
113	P46_2	GPIOINT112
114	P46_3	GPIOINT113
115	P47_0	GPIOINT114
116	P47_1	GPIOINT115
117	P47_2	GPIOINT116
118	P47_3	GPIOINT117
119	P48_0	GPIOINT118
120	P48_1	GPIOINT119
121	P48_2	GPIOINT120
122	P48_3	GPIOINT121
123	P48_4	GPIOINT122

8.7.4 Internal Interrupt

8.7.4.1 Bus Error Interrupt

Bus error interrupts (BUSERR_INT0-41) generated by system bus are integrated into one interrupt (BUS_ERR_INT) by IA55/IM33 and the integrated interrupt is notified to CPU.

[Status Control]

- Bus error interrupt requests can be confirmed by reading BUS Error Interrupt Status Control Register (BEISR0/1) in IA55/IM33.
  Refer to **Table 8.5** and **Table 8.6** for mapping BESTAT0-41 onto bus error interrupt causes.
- The interrupt requests can be cleared by writing 0 to BESTAT0-41 bit.

Table 8.5 BEISR0 Bus Error Mapping

BEISR0	Interrupt	Description
BESTAT0	BUSERR_INT0	Bus Write Error Interrupt for CA55
BESTAT1	BUSERR_INT1	Bus Read Error Interrupt for CA55
BESTAT2	BUSERR_INT2	Bus Write Error Interrupt for Mali-G31
BESTAT3	BUSERR_INT3	Bus Read Error Interrupt for Mali-G31
BESTAT4	BUSERR_INT4	Bus Write Error Interrupt for DMAC_S
BESTAT5	BUSERR_INT5	Bus Read Error Interrupt for DMAC_S
BESTAT6	BUSERR_INT6	Bus Write Error Interrupt for DMAC_NS
BESTAT7	BUSERR_INT7	Bus Read Error Interrupt for DMAC_NS
BESTAT8	BUSERR_INT8	Bus Write Error Interrupt for CST_ETR
BESTAT9	BUSERR_INT9	Bus Read Error Interrupt for CST_ETR
BESTAT10	BUSERR_INT10	Bus Write Error Interrupt for CST_AP
BESTAT11	BUSERR_INT11	Bus Read Error Interrupt for CST_AP
BESTAT12	BUSERR_INT12	Bus Write Error Interrupt for CM33_S
BESTAT13	BUSERR_INT13	Bus Read Error Interrupt for CM33_S
BESTAT14	BUSERR_INT14	Bus Write Error Interrupt for CM33_C
BESTAT15	BUSERR_INT15	Bus Read Error Interrupt for CM33_C
BESTAT16	BUSERR_INT16	Bus Write Error Interrupt for SDHI Ch0
BESTAT17	BUSERR_INT17	Bus Read Error Interrupt for SDHI Ch0
BESTAT18	BUSERR_INT18	Bus Write Error Interrupt for SDHI Ch1
BESTAT19	BUSERR_INT19	Bus Read Error Interrupt for SDHI Ch1
BESTAT20	BUSERR_INT20	Bus Write Error Interrupt for Gether Ch0
BESTAT21	BUSERR_INT21	Bus Read Error Interrupt for Gether Ch0
BESTAT22	BUSERR_INT22	Bus Write Error Interrupt for Gether Ch1
BESTAT23	BUSERR_INT23	Bus Read Error Interrupt for Gether Ch1
BESTAT24	BUSERR_INT24	Bus Write Error Interrupt for USB Ch0 Host
BESTAT25	BUSERR_INT25	Bus Read Error Interrupt for USB Ch0 Host
BESTAT26	BUSERR_INT26	Bus Write Error Interrupt for USB Ch1 Host
BESTAT27	BUSERR_INT27	Bus Read Error Interrupt for USB Ch1 Host
BESTAT28	BUSERR_INT28	Bus Write Error Interrupt for USB Ch0 Function
BESTAT29	BUSERR_INT29	Bus Read Error Interrupt for USB Ch0 Function
BESTAT30	BUSERR_INT30	Bus Write Error Interrupt for CRU Video
BESTAT31	BUSERR_INT31	Bus Read Error Interrupt for CRU Video

Table 8.6 BEISR1 Bus Error Mapping

BEISR1	Interrupt	Description
BESTAT32	BUSERR_INT32	Bus Write Error Interrupt for CRU Statistics
BESTAT33	BUSERR_INT33	Bus Read Error Interrupt for CRU Statistics
BESTAT34	BUSERR_INT34	Bus Write Error Interrupt for FCPVD
BESTAT35	BUSERR_INT35	Bus Read Error Interrupt for FCPVD
BESTAT36	BUSERR_INT36	Bus Write Error Interrupt for MIPI DSI LINK
BESTAT37	BUSERR_INT37	Bus Read Error Interrupt for MIPI DSI LINK
BESTAT38	BUSERR_INT38	Bus Write Error Interrupt for ISU
BESTAT39	BUSERR_INT39	Bus Read Error Interrupt for ISU
BESTAT40	BUSERR_INT40	Bus Write Error Interrupt for FCPCS
BESTAT41	BUSERR_INT41	Bus Read Error Interrupt for FCPCS
BESTAT42	—	Reserved
BESTAT43	—	Reserved
BESTAT44	—	Reserved

8.7.4.2 ECCRAM Error Interrupt

ECCRAM error interrupts generated by On-Chip RAM (ECCRAM0, ECCRAM1) are integrated into one interrupt from eight interrupts by IA55/IM33 and the integrated interrupt is notified to CPU.

Integrated 1bit error interrupt: EC7TIE1_0, EC7TIE1_1
Integrated 2bit error interrupt: EC7TIE2_0, EC7TIE2_1
Integrated overflow interrupt: EC7TIOVF_0, EC7TIOVF_1

[Status Control]

- ECCRAM error interrupt requests can be confirmed by reading ECCRAM Error Interrupt Status Control Register (EREISR0/1) in IA55/IM33.
  Refer to **Table 8.7** and **Table 8.8** for mapping E1STAT0-7, E2STAT0-7 and OFSTAT0-7 onto ECCRAM error interrupt causes.
- The interrupt requests can be cleared by writing 0 to E1STAT0-7, E2STAT0-7 and OFSTAT0-7 bit.

Table 8.7 ECCRAM0 ECC Error Mapping

EREISR0	Interrupt	Description	Integrated Interrupt
E1STAT0	ECCRAM0_1E0	ECC 1bit error interrupt bit[31:0]	EC7TIE1_0
E1STAT1	ECCRAM0_1E1	ECC 1bit error interrupt bit[63:32]	
E1STAT2	ECCRAM0_1E2	ECC 1bit error interrupt bit[95:64]	
E1STAT3	ECCRAM0_1E3	ECC 1bit error interrupt bit[127:96]	
E1STAT4	ECCRAM0_1E4	ECC 1bit error interrupt bit[159:128]	
E1STAT5	ECCRAM0_1E5	ECC 1bit error interrupt bit[191:160]	
E1STAT6	ECCRAM0_1E6	ECC 1bit error interrupt bit[223:192]	
E1STAT7	ECCRAM0_1E7	ECC 1bit error interrupt bit[255:224]	
E2STAT0	ECCRAM0_2E0	ECC 2bit error interrupt bit[31:0]	EC7TIE2_0
E2STAT1	ECCRAM0_2E1	ECC 2bit error interrupt bit[63:32]	
E2STAT2	ECCRAM0_2E2	ECC 2bit error interrupt bit[95:64]	
E2STAT3	ECCRAM0_2E3	ECC 2bit error interrupt bit[127:96]	
E2STAT4	ECCRAM0_2E4	ECC 2bit error interrupt bit[159:128]	
E2STAT5	ECCRAM0_2E5	ECC 2bit error interrupt bit[191:160]	
E2STAT6	ECCRAM0_2E6	ECC 2bit error interrupt bit[223:192]	
E2STAT7	ECCRAM0_2E7	ECC 2bit error interrupt bit[255:224]	
OFSTAT0	ECCRAM0_OF0	ECC error overflow interrupt bit[31:0]	EC7TIOVF_0
OFSTAT1	ECCRAM0_OF1	ECC error overflow interrupt bit[63:32]	
OFSTAT2	ECCRAM0_OF2	ECC error overflow interrupt bit[95:64]	
OFSTAT3	ECCRAM0_OF3	ECC error overflow interrupt bit[127:96]	
OFSTAT4	ECCRAM0_OF4	ECC error overflow interrupt bit[159:128]	
OFSTAT5	ECCRAM0_OF5	ECC error overflow interrupt bit[191:160]	
OFSTAT6	ECCRAM0_OF6	ECC error overflow interrupt bit[223:192]	
OFSTAT7	ECCRAM0_OF7	ECC error overflow interrupt bit[255:224]	

Table 8.8 ECCRAM1 ECC Error Mapping

EREISR1	Interrupt	Description	Integrated Interrupt
E1STAT0	ECCRAM1_1E0	ECC 1bit error interrupt bit[31:0]	EC7TIE1_1
E1STAT1	ECCRAM1_1E1	ECC 1bit error interrupt bit[63:32]	
E1STAT2	ECCRAM1_1E2	ECC 1bit error interrupt bit[95:64]	
E1STAT3	ECCRAM1_1E3	ECC 1bit error interrupt bit[127:96]	
E1STAT4	ECCRAM1_1E4	ECC 1bit error interrupt bit[159:128]	
E1STAT5	ECCRAM1_1E5	ECC 1bit error interrupt bit[191:160]	
E1STAT6	ECCRAM1_1E6	ECC 1bit error interrupt bit[223:192]	
E1STAT7	ECCRAM1_1E7	ECC 1bit error interrupt bit[255:224]	
E2STAT0	ECCRAM1_2E0	ECC 2bit error interrupt bit[31:0]	EC7TIE2_1
E2STAT1	ECCRAM1_2E1	ECC 2bit error interrupt bit[63:32]	
E2STAT2	ECCRAM1_2E2	ECC 2bit error interrupt bit[95:64]	
E2STAT3	ECCRAM1_2E3	ECC 2bit error interrupt bit[127:96]	
E2STAT4	ECCRAM1_2E4	ECC 2bit error interrupt bit[159:128]	
E2STAT5	ECCRAM1_2E5	ECC 2bit error interrupt bit[191:160]	
E2STAT6	ECCRAM1_2E6	ECC 2bit error interrupt bit[223:192]	
E2STAT7	ECCRAM1_2E7	ECC 2bit error interrupt bit[255:224]	
OFSTAT0	ECCRAM1_OF0	ECC error overflow interrupt bit[31:0]	EC7TIOVF_1
OFSTAT1	ECCRAM1_OF1	ECC error overflow interrupt bit[63:32]	
OFSTAT2	ECCRAM1_OF2	ECC error overflow interrupt bit[95:64]	
OFSTAT3	ECCRAM1_OF3	ECC error overflow interrupt bit[127:96]	
OFSTAT4	ECCRAM1_OF4	ECC error overflow interrupt bit[159:128]	
OFSTAT5	ECCRAM1_OF5	ECC error overflow interrupt bit[191:160]	
OFSTAT6	ECCRAM1_OF6	ECC error overflow interrupt bit[223:192]	
OFSTAT7	ECCRAM1_OF7	ECC error overflow interrupt bit[255:224]	

8.8 Usage Note

8.8.1 Precaution when use the peripheral modules which can initiate DMA Controller.

Some on-chip peripheral modules use the same signal both for an interrupt request and for a DMA transfer request.
Refer to **Section 14.4.25, DMA Extension Resource Selectors 0/0S to 7/7S** for detail.

If such a module is selected by a DMARSn/nS register, the signal works as a DMA transfer request signal and interrupt requests to the interrupt controller are masked.

To enable the interrupt, clear the setting of DMARSn/nS (set all MID[7:0] and RID[1:0] to 0).
In addition, need following register settings even if DMA controller is not used.

- Set CPG_CLKON_DMAC_REG register to supply a clock for DMA Controller.
  Refer to **Section 7.2.4, Clock Control Register DMAC_REG** for register detail.
- Set CPG_RST_DMAC register to release a reset for DMA Controller.
  Refer to **Section 7.2.4, Reset Control Register DMAC** for register detail.

8.8.2 Clear Timing of Interrupt Cause

Clear the interrupt cause flag in the interrupt exception service routine. It takes time from clearing the interrupt cause flag until the interrupt cause to the CPU is actually removed. Therefore, to prevent the interrupt cause that should have been cleared from being accidentally accepted again, the interrupt cause flag is read after clearing, and then the return instruction is executed.

8.8.3 Precaution when Changing Interrupt Settings

To change the NMI, IRQ, TINT interrupt settings, mask the interrupts and change the settings.

- When changing the noise filter settings.
- When switching the GPIO pins to IRQ or GPIOINT.
- When changing the source of TINT.
- When changing the interrupt detection method.

When changing the NMI, IRQ, TINT interrupt detection method to the edge type, perform the process of clearing the interrupt status after changing the setting.

- NMI: Write 0 to the NSTAT bit of NSCR.
- IRQ: Write 0 to the ISTATn bit of ISCR.
- TINT: Write 0 to the TSTATn bit of TSCR.

9. DDR3L/DDR4 SDRAM Memory Controller (MEMC)

DDR3L/DDR4 SDRAM Memory Controller (MEMC) is External Bus Controller for DDR3L/DDR4 SDRAM (DDR). This block supports up to DDR4-1600 and DDR3L-1333 SDRAM. Interface bus width is 16 bits. In line ECC can be supported.
For sepup and configuration to this block, please use and refer to the linux software package of this LSI including DDR setup codes.

9.1 Features

This unit consists of MC (Memory controller) and PHY. It supports the following specifications.

- DDR3L (JEDEC STANDARD JESD79-3F, JESD79-3-1A.01)
- DDR4 (JEDEC STANDARD JESD79-4C)

Figure 9.1 shows features of this block.

Table 9.1 DDR3L/DDR4 SDRAM Memory Controller (MEMC) features

Feature	Description
DRAM IF	● DDR3L: 800 to 1333 Mbps (400 to 666 MHz) ● DDR4: 1333 to 1600 Mbps (666 to 800 MHz) ● Width: 16 bits (Full (16 bits) or Half (8 bits) datawidth mode) ● Rank: 1, 2
MC (Memory controller)	● Fully pipelined command, read and write data interfaces to the controller. ● Advanced bank look-ahead features for high memory throughput. ● A programmable register interface to control memory parameters and protocols including auto pre-charge. ● Full initialization of memory on controller reset. ● ECC function for single bit and double bit error reporting, single bit error correction, and programmable removal of ECC storage.
PHY	● Write Leveling (Deskew CK vs Write-DQS) ● Bit Leveling (Deskew DQS vs DQ/DM, Read and Write) ● Vref Training (Read and Write) ● Self Calibration Logic (Deskew Read vs Internal clock) ● Command/Address Swizzling

The diagrams of this unit are follows.

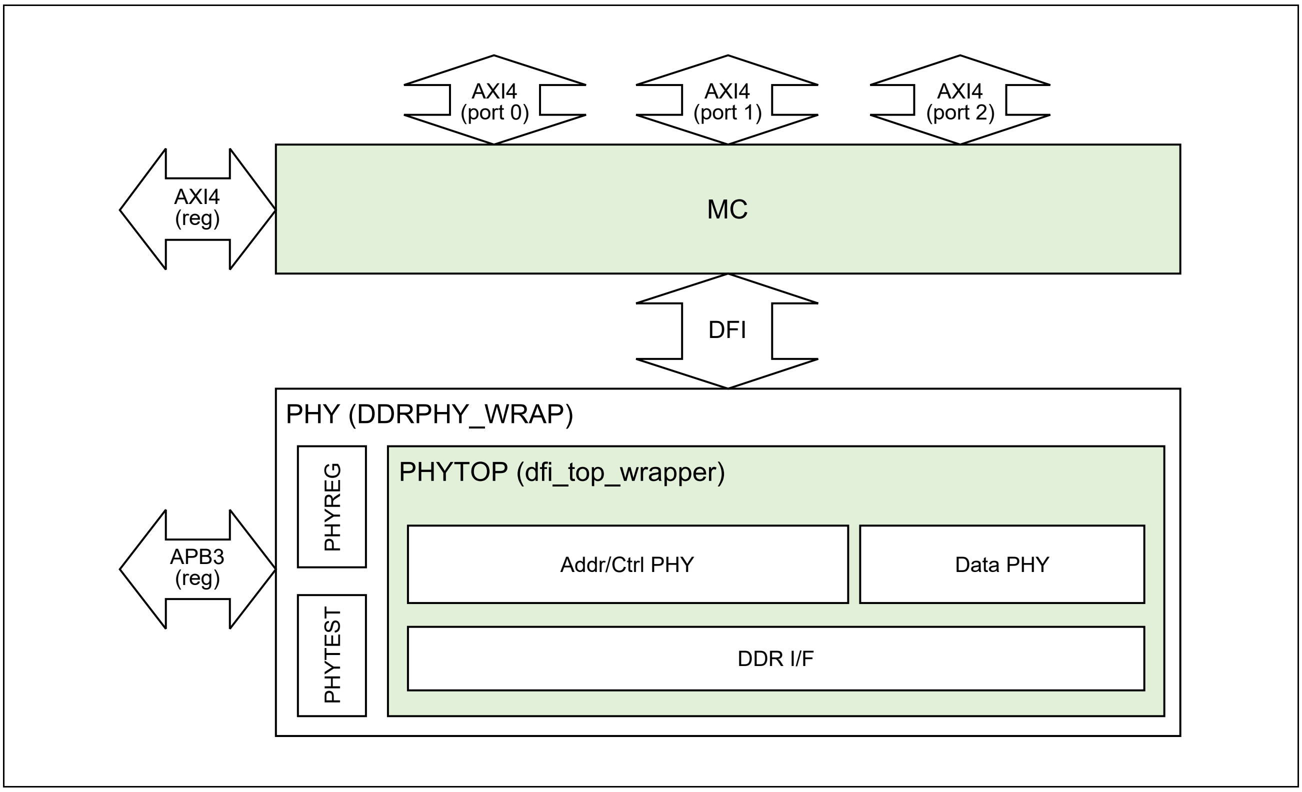


Figure 9.1 Block Diagram

This unit has the following restrictions.

DDR3L

- BC4 is not supported.
- Interleave is not supported.

DDR4

- BC4 is not supported.
- Interleave is not supported.
- Geardown Mode is not supported.
- Fine granularity refresh mode is supported, but only in Fixed x1 mode.
- CAL Mode is not supported.
- DBI is not supported.
- CRC is not supported.
- CA Parity is not supported.

9.2 Interface

9.2.1 Power, Ground

Table 9.2 shows the power and ground pins of this unit.

Table 9.2 Power, Ground

Name	I/O	Description
DDR_VDDQ	Supply	Power supply for I/O
VDD	Supply	Power supply for core
VSS	Ground	Ground for core

9.2.2 External Pins

Table 9.3 shows the external pins for DDR interface.

Table 9.3 External pin list

External pin	I/O	Description
DDR_CLK_P	O	DDR Clock (Pos)
DDR_CLK_N	O	DDR Clock (Neg)
DDR_RESET#	O	DDR Active Low Asynchronous Reset
DDR_CKE	O	DDR Clock Enable
DDR_CS0#	O	DDR Address or Command
DDR_CS1#	O	DDR Address or Command
DDR_RAS#	O	DDR Address or Command
DDR_CAS#	O	DDR Address or Command
DDR_WE#	O	DDR Address or Command
DDR_BA0 to DDR_BA2	O	DDR Address or Command
DDR_ADDR0 to DDR_ADDR15	O	DDR Address or Command
DDR_ODT0	O	DDR On Die Termination
DDR_ODT1	O	DDR On Die Termination
DDR_DQS0_P	IO	DDR Data Strobe (Pos)
DDR_DQS0_N	IO	DDR Data Strobe (Neg)
DDR_DQS1_P	IO	DDR Data Strobe (Pos)
DDR_DQS1_N	IO	DDR Data Strobe (Neg)
DDR_DQ0 to DDR_DQ15	IO	DDR Data
DDR_DM0	O	DDR Data Mask
DDR_DM1	O	DDR Data Mask
DDR_CALIBRATION	IO	Should be connected to an external 240-ohm resistor. It is used for drive and termination impedance calibration of IOs.

9.3 Usage Note

This unit has the following restrictions.

Table 9.4 External pin list

Item	Restriction
Address map	It has the area (address) that access is prohibited, because it is used by the periodic training. It is as below. DRAM address: BA = 0, ROW = H'0, COL = H'000 to H'00F BG = 0 and 1 (only DDR4) System address: H'040000000 to H'04000003F (if DDR4) H'040000000 to H'04000001F (if DDR3L)
Period of DRAM inaccessibility	It has the period of DRAM inaccessibility to execute the periodic training. It depends only on the Bit Rate, and not on the kind of DRAM. It is belows: 1600Mbps: 5.0 μsec every 1000 msec 1333Mbps: 1.5 μsec every 1000 msec 1066Mbps: 1.875 μsec every 1000 msec 800Mbps: 2.5 μsec every 1000 msec

9.3.1 Power On

The power on sequence for the DDR_VDDQ and VDD domains is such that either the 2 voltages should be ramped up together or VDD should be ramped up prior to DDR_VDDQ in order to prevent voltage stress in the DDR IOs.

10. On-chip RAM

This LSI has an on-chip RAM for work area. These memory units can be used to store instructions or data. The operation and write access to the on-chip RAM can be enabled or disabled through the RAM enable bit (VCEN in SYS_RAMn_EN register) (n = 0-1) and RAM write enable bit (VLWEN in SYS_RAMn_EN register) (n = 0-1).

For details on the register, refer to the **Section 6.3** in chapter SYSC.

10.1 Features

■ Memory size

128 Kbytes (SRAM ACPU: 64 Kbytes, SRAM MCPU: 64 Kbytes)

■ Ports

On-chip RAM has one read and write port and is connected to the AXI bus.

■ Method of arbitration

When the same port of the on-chip RAM is accessed from different masters simultaneously, the AXI bus performs arbitration in round-robin mode.

■ Number of access cycles

The number of cycles for access to read or write is one cycle of SRAM_ACPU_ACLK (P1ϕ).

The number of cycles for access to read or write is one cycle of SRAM_MCPU_ACLK (I2ϕ).

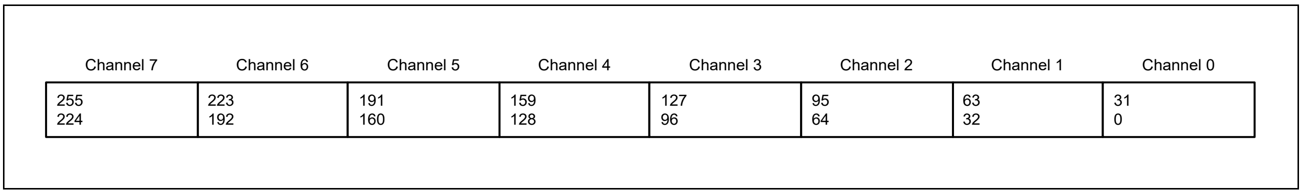
11. Error Correcting Code (ECC) for On-Chip RAM

11.1 Overview

This LSI chip has two 64-Kbyte areas of on-chip RAM for use as the working area.

Each of on-chip RAM incorporates an ECC function, which allows the detection and correction of 1-bit errors and the detection of 2-bit errors. When a 1-bit or 2-bit error occurs, the address where the error occurred can be stored in an ECC error address register.

The data bus width is 256 bits. This is divided into eight channels as shown in **Figure 11.1**, with detection handled in each 32-bit channel.



Channel 7	Channel 6	Channel 5	Channel 4	Channel 3	Channel 2	Channel 1	Channel 0
255 224	223 192	191 160	159 128	127 96	95 64	63 32	31 0

Figure 11.1 Division for ECC Processing

11.1.1 Features

- Error detection and correction
  - Detection and correction of 1-bit errors and detection of 2-bit errors are possible.
  - An error involving 3 or more bits cannot be correctly handled and may result in erroneous detection and correction.
- Control of error detection and correction
  - Error detection and correction, and correction of 1-bit errors can respectively be enabled or disabled.
- Capture of the addresses where errors occurred
  - When a 1-bit or 2-bit error is detected, the address where the error was encountered is stored in an ECC error address register. Each channel has eight ECC error address registers.
- Interrupt request
  - An interrupt request can be generated on the detection of a 1-bit error (selectable as enabled or disabled).
  - An interrupt request can be generated on the detection of a 2-bit error (selectable as enabled or disabled).
  - An overflow interrupt is generated from each channel on detection of an error when all ECC error address registers contain values.

11.2 Register Configuration

Table 11.1 and **Table 11.2** respectively show the base register and base address. The addresses of the ECC registers are represented by offsets from the base address.

Table 11.1 Base Register

Base Register Name	On-Chip RAM	Register Abbreviation
<REG_base>	SRAM ACPU	ECCRAM0
	SRAM MCPU	ECCRAM1

Table 11.2 Base Register

Base Register Name	On-Chip RAM	Register Abbreviation
<ADR_base>	SRAM ACPU	H’0_1108_0000 (Overall Address Space) H’4108_0000 (Cortex-M33 Address Space Non-Secure) H’5108_0000 (Cortex-M33 Address Space Secure)
	SRAM MCPU	H’0_1109_0000 (Overall Address Space) H’4109_0000 (Cortex-M33 Address Space Non-Secure) H’5109_0000 (Cortex-M33 Address Space Secure)

Table 11.3 lists the ECC registers.

Do not attempt access to the on-chip RAM areas while the registers listed in **Table 11.3** are being set.

Table 11.3 List of the ECC Registers (1/2)

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	ECC control register_0	<REG_base>CTL_0	R/W	H'0000_0010	<ADR_base> + H'0000	32
	ECC error address register 0_0	<REG_base>EAD0_0	R	H'0000_0000	<ADR_base> + H'0010	32
	ECC error address register 1_0	<REG_base>EAD1_0	R	H'0000_0000	<ADR_base> + H'0014	32
	ECC error address register 2_0	<REG_base>EAD2_0	R	H'0000_0000	<ADR_base> + H'0018	32
	ECC error address register 3_0	<REG_base>EAD3_0	R	H'0000_0000	<ADR_base> + H'001C	32
	ECC error address register 4_0	<REG_base>EAD4_0	R	H'0000_0000	<ADR_base> + H'0020	32
	ECC error address register 5_0	<REG_base>EAD5_0	R	H'0000_0000	<ADR_base> + H'0024	32
	ECC error address register 6_0	<REG_base>EAD6_0	R	H'0000_0000	<ADR_base> + H'0028	32
	ECC error address register 7_0	<REG_base>EAD7_0	R	H'0000_0000	<ADR_base> + H'002C	32
1	ECC control register_1	<REG_base>CTL_1	R/W	H'0000_0010	<ADR_base> + H'0040	32
	ECC error address register 0_1	<REG_base>EAD0_1	R	H'0000_0000	<ADR_base> + H'0050	32
	ECC error address register 1_1	<REG_base>EAD1_1	R	H'0000_0000	<ADR_base> + H'0054	32
	ECC error address register 2_1	<REG_base>EAD2_1	R	H'0000_0000	<ADR_base> + H'0058	32
	ECC error address register 3_1	<REG_base>EAD3_1	R	H'0000_0000	<ADR_base> + H'005C	32
	ECC error address register 4_1	<REG_base>EAD4_1	R	H'0000_0000	<ADR_base> + H'0060	32
	ECC error address register 5_1	<REG_base>EAD5_1	R	H'0000_0000	<ADR_base> + H'0064	32
	ECC error address register 6_1	<REG_base>EAD6_1	R	H'0000_0000	<ADR_base> + H'0068	32
	ECC error address register 7_1	<REG_base>EAD7_1	R	H'0000_0000	<ADR_base> + H'006C	32
2	ECC control register_2	<REG_base>CTL_2	R/W	H'0000_0010	<ADR_base> + H'0080	32
	ECC error address register 0_2	<REG_base>EAD0_2	R	H'0000_0000	<ADR_base> + H'0090	32
	ECC error address register 1_2	<REG_base>EAD1_2	R	H'0000_0000	<ADR_base> + H'0094	32
	ECC error address register 2_2	<REG_base>EAD2_2	R	H'0000_0000	<ADR_base> + H'0098	32
	ECC error address register 3_2	<REG_base>EAD3_2	R	H'0000_0000	<ADR_base> + H'009C	32
	ECC error address register 4_2	<REG_base>EAD4_2	R	H'0000_0000	<ADR_base> + H'00A0	32
	ECC error address register 5_2	<REG_base>EAD5_2	R	H'0000_0000	<ADR_base> + H'00A4	32
	ECC error address register 6_2	<REG_base>EAD6_2	R	H'0000_0000	<ADR_base> + H'00A8	32
	ECC error address register 7_2	<REG_base>EAD7_2	R	H'0000_0000	<ADR_base> + H'00AC	32
3	ECC control register_3	<REG_base>CTL_3	R/W	H'0000_0010	<ADR_base> + H'00C0	32
	ECC error address register 0_3	<REG_base>EAD0_3	R	H'0000_0000	<ADR_base> + H'00D0	32
	ECC error address register 1_3	<REG_base>EAD1_3	R	H'0000_0000	<ADR_base> + H'00D4	32
	ECC error address register 2_3	<REG_base>EAD2_3	R	H'0000_0000	<ADR_base> + H'00D8	32
	ECC error address register 3_3	<REG_base>EAD3_3	R	H'0000_0000	<ADR_base> + H'00DC	32
	ECC error address register 4_3	<REG_base>EAD4_3	R	H'0000_0000	<ADR_base> + H'00E0	32
	ECC error address register 5_3	<REG_base>EAD5_3	R	H'0000_0000	<ADR_base> + H'00E4	32
	ECC error address register 6_3	<REG_base>EAD6_3	R	H'0000_0000	<ADR_base> + H'00E8	32
	ECC error address register 7_3	<REG_base>EAD7_3	R	H'0000_0000	<ADR_base> + H'00EC	32

Table 11.3 List of the ECC Registers (2/2)

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
4	ECC control register_4	<REG_base>CTL_4	R/W	H’0000_0010	<ADR_base> + H’0100	32
	ECC error address register 0_4	<REG_base>EAD0_4	R	H’0000_0000	<ADR_base> + H’0110	32
	ECC error address register 1_4	<REG_base>EAD1_4	R	H’0000_0000	<ADR_base> + H’0114	32
	ECC error address register 2_4	<REG_base>EAD2_4	R	H’0000_0000	<ADR_base> + H’0118	32
	ECC error address register 3_4	<REG_base>EAD3_4	R	H’0000_0000	<ADR_base> + H’011C	32
	ECC error address register 4_4	<REG_base>EAD4_4	R	H’0000_0000	<ADR_base> + H’0120	32
	ECC error address register 5_4	<REG_base>EAD5_4	R	H’0000_0000	<ADR_base> + H’0124	32
	ECC error address register 6_4	<REG_base>EAD6_4	R	H’0000_0000	<ADR_base> + H’0128	32
	ECC error address register 7_4	<REG_base>EAD7_4	R	H’0000_0000	<ADR_base> + H’012C	32
5	ECC control register_5	<REG_base>CTL_5	R/W	H’0000_0010	<ADR_base> + H’0140	32
	ECC error address register 0_5	<REG_base>EAD0_5	R	H’0000_0000	<ADR_base> + H’0150	32
	ECC error address register 1_5	<REG_base>EAD1_5	R	H’0000_0000	<ADR_base> + H’0154	32
	ECC error address register 2_5	<REG_base>EAD2_5	R	H’0000_0000	<ADR_base> + H’0158	32
	ECC error address register 3_5	<REG_base>EAD3_5	R	H’0000_0000	<ADR_base> + H’015C	32
	ECC error address register 4_5	<REG_base>EAD4_5	R	H’0000_0000	<ADR_base> + H’0160	32
	ECC error address register 5_5	<REG_base>EAD5_5	R	H’0000_0000	<ADR_base> + H’0164	32
	ECC error address register 6_5	<REG_base>EAD6_5	R	H’0000_0000	<ADR_base> + H’0168	32
	ECC error address register 7_5	<REG_base>EAD7_5	R	H’0000_0000	<ADR_base> + H’016C	32
6	ECC control register_6	<REG_base>CTL_6	R/W	H’0000_0010	<ADR_base> + H’0180	32
	ECC error address register 0_6	<REG_base>EAD0_6	R	H’0000_0000	<ADR_base> + H’0190	32
	ECC error address register 1_6	<REG_base>EAD1_6	R	H’0000_0000	<ADR_base> + H’0194	32
	ECC error address register 2_6	<REG_base>EAD2_6	R	H’0000_0000	<ADR_base> + H’0198	32
	ECC error address register 3_6	<REG_base>EAD3_6	R	H’0000_0000	<ADR_base> + H’019C	32
	ECC error address register 4_6	<REG_base>EAD4_6	R	H’0000_0000	<ADR_base> + H’01A0	32
	ECC error address register 5_6	<REG_base>EAD5_6	R	H’0000_0000	<ADR_base> + H’01A4	32
	ECC error address register 6_6	<REG_base>EAD6_6	R	H’0000_0000	<ADR_base> + H’01A8	32
	ECC error address register 7_6	<REG_base>EAD7_6	R	H’0000_0000	<ADR_base> + H’01AC	32
7	ECC control register_7	<REG_base>CTL_7	R/W	H’0000_0010	<ADR_base> + H’01C0	32
	ECC error address register 0_7	<REG_base>EAD0_7	R	H’0000_0000	<ADR_base> + H’01D0	32
	ECC error address register 1_7	<REG_base>EAD1_7	R	H’0000_0000	<ADR_base> + H’01D4	32
	ECC error address register 2_7	<REG_base>EAD2_7	R	H’0000_0000	<ADR_base> + H’01D8	32
	ECC error address register 3_7	<REG_base>EAD3_7	R	H’0000_0000	<ADR_base> + H’01DC	32
	ECC error address register 4_7	<REG_base>EAD4_7	R	H’0000_0000	<ADR_base> + H’01E0	32
	ECC error address register 5_7	<REG_base>EAD5_7	R	H’0000_0000	<ADR_base> + H’01E4	32
	ECC error address register 6_7	<REG_base>EAD6_7	R	H’0000_0000	<ADR_base> + H’01E8	32
	ECC error address register 7_7	<REG_base>EAD7_7	R	H’0000_0000	<ADR_base> + H’01EC	32

11.3 Register Descriptions

11.3.1 ECC Control Registers_n (<REG_base>CTL_n) (n = 0 to 7)

Each ECC control register_n is used to control the ECC function in the corresponding channel n shown in **Figure 11.1**.

After writing the initial value to the entire usage area of the on-chip RAM, set the given ECERVF bits to 1 to enable error detection. Be sure to write 01b to the corresponding EMCA[1:0] bits to enable writing to an ECERVF bit.

For writing the initial value to the on-chip RAM areas, refer to **Section 11.5, Initializing the Detection-Usage Areas of the On-Chip RAM**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECDEDF7	ECSEDF7	ECDEDF6	ECSEDF6	ECDEDF5	ECSEDF5	ECDEDF4	ECSEDF4	ECDEDF3	ECSEDF3	ECDEDF2	ECSEDF2	ECDEDF1	ECSEDF1	ECDEDF0	ECSEDF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EMCA[1:0]		—	—	ECOVFF	ECER2C	ECER1C	—	—	ECERVF	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	ECDEDF7	0	R	ECC 2-Bit Error Detection Flags ECDEDFm (m = 0 to 7) These bits indicate the occurrence of 2-bit errors in the data output from the RAM while error detection is enabled. The addresses of the errors will be stored in the <REG_base>EADm_n registers. ECC 1-Bit Error Detection Flags ECSEDFm (m = 0 to 7) These bits indicate the occurrence of 1-bit errors in the data output from the RAM while error detection is enabled. The addresses of the errors will be stored in the <REG_base>EADm_n registers. The ECDEDFm and ECSEDFm bits are cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER2C bit 3. Setting the ECC error detection enable bit to “disabled” (ECERVF = 0) 4. Reset by WDT overflow
30	ECSEDF7	0	R	
29	ECDEDF6	0	R	
28	ECSEDF6	0	R	
27	ECDEDF5	0	R	
26	ECSEDF5	0	R	
25	ECDEDF4	0	R	
24	ECSEDF4	0	R	
23	ECDEDF3	0	R	
22	ECSEDF3	0	R	
21	ECDEDF2	0	R	
20	ECSEDF2	0	R	
19	ECDEDF1	0	R	
18	ECSEDF1	0	R	
17	ECDEDF0	0	R	
16	ECSEDF0	0	R	
15, 14	EMCA[1:0]	00b	R/W	Enable Writing to ECERVF 01b: Enabled Others: Disabled These bits are always read as 00b.
13, 12	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
11	ECOVFF	0	R	ECC Overflow Detection Flag This bit is set to 1 and an overflow interrupt is output on detection of an error when the ECC error address registers (<REG_base>EADm_n (m = 0 to 7)) all contain data. An overflow interrupt is output even if the error is detected while the setting of this bit is already 1. This bit is cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER2C bit or ECER1C bit 3. Setting the ECC error detection enable bit to “disabled” (ECERVF = 0)
10	ECER2C	0	R/W	2-Bit Error Detection Flag Clear 0: The internal state does not change. 1: The corresponding 2-bit error detection flag ECER2F is cleared. This bit is always read as 0.
9	ECER1C	0	R/W	1-Bit Error Detection Flag Clear 0: The internal state does not change. 1: The corresponding 1-bit error detection flag ECER1F is cleared. This bit is always read as 0.
8, 7	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	ECERVF	0	R/W	ECC Error Detection Enable 0: Disables error detection. 1: Enables error detection. 1-bit error correction (EC1ECP) and interrupt control (EC2EDIC and EC1EDIC) are only possible when error detection is enabled. *Note*: To write to this bit, the corresponding EMCA[1:0] bits must be set to 01b.
5	EC1ECP	0	R/W	1-Bit Error Correction Enable 0: The error is corrected when a 1-bit error is detected. 1: The error is not corrected when a 1-bit error is detected.
4	EC2EDIC	1	R/W	2-Bit Error Detection Interrupt Control 0: An interrupt is not generated when a 2-bit error is detected. 1: An interrupt is generated when a 2-bit error is detected. The interrupt is only generated when the corresponding 2-bit error detection flag (ECER2F) changes from 0 to 1.
3	EC1EDIC	0	R/W	1-Bit Error Detection Interrupt Control 0: An interrupt is not generated when a 1-bit error is detected. 1: An interrupt is generated when a 1-bit error is detected. The interrupt is only generated when the corresponding 1-bit error detection flag (ECER1F) changes from 0 to 1.
2	ECER2F	0	R	2-Bit Error Detection Flag 0: A 2-bit error has not occurred. 1: A 2-bit error has occurred. This bit is cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER2C bit 3. Setting the ECC error detection enable bit to "disabled" (ECERVF = 0)
1	ECER1F	0	R	1-Bit Error Detection Flag 0: A 1-bit error has not occurred. 1: A 1-bit error has occurred. This bit is cleared to 0 under any of the following conditions. 1. Power-on reset 2. Writing 1 to the corresponding ECER1C bit 3. Setting the ECC error detection enable bit to "disabled" (ECERVF = 0)

Bit	Bit Name	Initial Value	R/W	Description
0	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

11.3.2 ECC Error Address Registers m_n (<REG_base>EADm_n) (m = 0 to 7) (n = 0 to 7)

When a 1- or 2-bit error occurs, up to eight error addresses are sequentially stored in the registers from <REG_base>EAD0_n to <REG_base>EAD7_n.

When a 2-bit error is detected while only 1-bit error addresses were stored up to <REG_base>EAD7_n, <REG_base>EAD7_n is overwritten by the address of the 2-bit error. Otherwise, it will not be overwritten. For details, refer to **Section 11.6.4, ECC Error Address Register Overwrite Conditions**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECEAD0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECEAD0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ECEAD0 [31:0]	All 0	R	These bits hold the address of the (m + 1)th error to have been detected. These bits are cleared to 0 under the following condition: clearing ECER2F or ECER1F in the <REG_base>CTL_n register.

11.4 Initializing the ECC Function

Initialize the ECC by following the procedure below to enable the ECC function.

1. Set the VECCEN bit in the ECCRAM setting register corresponding to the RAM area for which the ECC function is to be enabled to 1.
2. Write the initial value to the entire usage area of the RAM for which the ECC function is to be enabled.
3. Set the ECERVF bits in the ECC control registers_n to 1 to enable error detection. Then, set the respective bits to control 1-bit error correction and interrupts.

For notes on the settings of registers of this module, refer to **Section 11.6.1, Notes on Setting the Registers of This Module**.

11.5 Initializing the Detection-Usage Areas of the On-Chip RAM

Before using the ECC function, write the initial value to the entire usage area of the on-chip RAM for which the ECC function is to be enabled. The initial value can be any value. Since the detection of ECC errors is handled in 32-bit units in each channel, the initial value should also be written in 32-bit units.

Enabling error detection without initialization may cause an ECC error due to unintended reading.

Examples of Operations that May Lead to Errors

1. When writing is to proceed in 8- or 16-bit units, reading the 32-bit value from the given address and then writing the data
2. If the locations are not initialized, an ECC error may occur when the 32-bit value is read. For details on writing in 8- or 16-bit units, refer to **section 11.6.2, 8- or 16-Bit Access**.
3. When the CPU cache is enabled, if a cache miss occurs in writing, the cache being refilled by a 1-line unit
   At this time if data are not initialized, an ECC error may occur because data are read in 1-line (32-byte) units.

11.6 Usage Notes

11.6.1 Notes on Setting the Registers of This Module

Ensure that none of the bus masters proceeds with access to the on-chip RAM areas while the registers listed in **Table 11.3** are being set up.

11.6.2 8- or 16-Bit Access

The ECC function for the on-chip RAM areas handles error detection in 32-bit units. Accordingly, when access is to be in 8- or 16-bit units, the operations proceed as follows.

(1) 8- or 16-bit reading

(a) The 32 bits of data are read and the ECC is used to determine whether or not an error is present.

(b) The 8 or 16 bits of data are then returned to the bus master that issued the read request.

(2) 8- or 16-bit writing

(a) The 32 bits of data from the address that contains the location for writing are read and the ECC is used to determine whether or not an error is present.

(b) The 32 bits of data that have been read are then overwritten with the 8- or 16-bit data.

(c) The ECC value for the overwritten 32-bit data in (b) is added by writing it to the RAM.

11.6.3 Numbers of Cycles for Reading from or Writing to a RAM Area with the ECC Function Enabled

The numbers of cycles for reading from or writing to an on-chip RAM area with the ECC function enabled are as follows.

(1) Reading

The number of cycles for reading in 8- or 16-bit and 32-bit units when the ECC function is enabled is one cycle of SRAM_ACPU_ACLK or SRAM_MCPU_ACLK longer than when the ECC function is disabled.

(2) Writing

The number of cycles for writing in 8- or 16-bit units when the ECC function is enabled is two cycles of SRAM_ACPU_ACLK or SRAM_MCPU_ACLK longer than when the ECC function is disabled. The number of cycles for writing in 32-bit units when the ECC function is enabled is the same as when the ECC function is disabled.

11.6.4 ECC Error Address Register Overwrite Conditions

If a new ECC error occurs with the ECC error address stored up to <REG_base> EAD7_n, <REG_base> EAD7_n may or may not be overwritten. The conditions to be overwritten are shown below.

Error Address Storage Status Up to <REG_base> EAD7_n	New ECC Error	<REG_base>EAD7_n Overwrite	Overflow Interrupt
Only 1-bit error	1-bit error	Not overwritten	Occur
Only 1-bit error	2-bit error	Overwritten	Occur
Only 2-bit error	1-bit error	Not overwritten	Occur
Only 2-bit error	2-bit error	Not overwritten	Occur
1-bit error and 2-bit error	1-bit error	Not overwritten	Occur
1-bit error and 2-bit error	2-bit error	Not overwritten	Occur

12. Message Handling Unit (MHU)

MHU is a function for message communication between Cortex-A55(CA55) cores and Cortex-M33(CM33). Message communication is done by shared RAM (On-chip RAM) for passing message and response between CPUs and the function (MHU) for notifying when messages and responses are stored in the memory.

12.1 Features

Table 12.1 shows MHU feature summary.

Table 12.1 Feature summary

Feature	Description
Interrupt by Message and response	Generate interrupt to notify each Cortex-A55 or Cortex-M33 when the message/response is stored in shared RAM (On-chip RAM).
Software interrupt control	Control registers for interrupt setting, clearing, and status confirmation

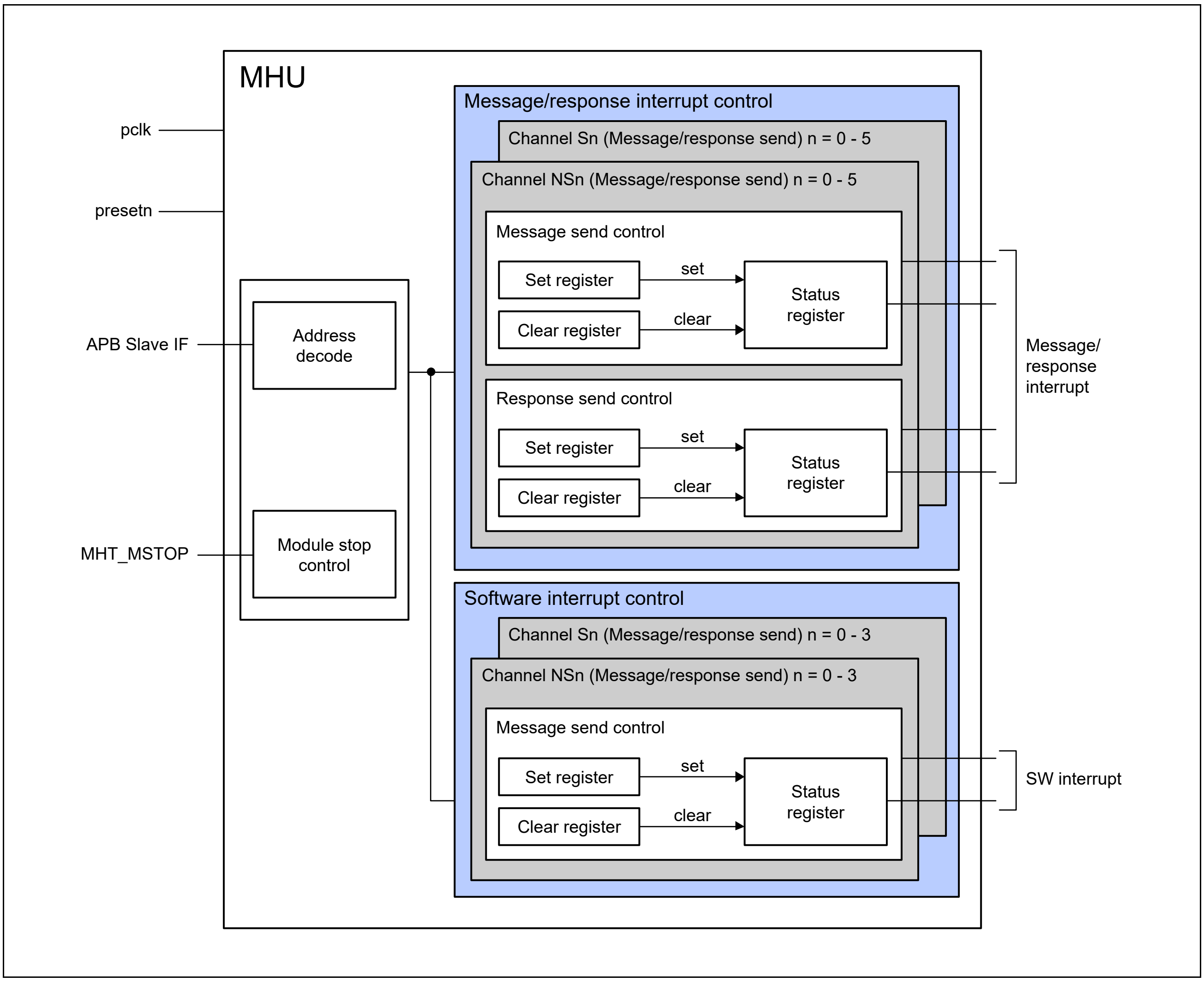


Figure 12.1 MHU Block diagram

Message/response interrupt control registers

Registers that control message/response interrupts.

The message/response interrupt control register controls interrupts with the set register, clear register, and status register. An interrupt is asserted by writing “1” to the set register. The interrupt is negated by writing “1” to the clear register. The status can be checked by reading the status register. With these as one set, the message transmission processing register

One channel is composed of a pair of data transmission processing register and response transmission processing register, and a total of 12 channels are mounted. The breakdown of 12 channels is as follows.

Shows MHU feature summary.

Non-Secure interrupt control register(Channel NS0-5):

Channel NS0 (CA55 Core0: Message transmission/CA55 Core1: Response transmission)

Channel NS1 (CA55 Core0: Message transmission/CM33: Response transmission)

Channel NS2 (CA55 Core1: Message transmission/CA55 Core0: Response transmission)

Channel NS3 (CA55 Core1: Message transmission/CM33: Response transmission)

Channel NS4 (CM33: Message transmission/CA55 Core0: Response transmission)

Channel NS5 (CM33: Message transmission/CA55 Core1: Response transmission)

Secure interrupt control register(Channel S0-5):

Channel S0 (CA55 Core0: Message transmission/CA55 Core1: Response transmission)

Channel S1 (CA55 Core0: Message transmission/CM33 : Response transmission)

Channel S2 (CA55 Core1: Message transmission/CA55 Core0: Response transmission)

Channel S3 (CA55 Core1: Message transmission/CM33: Response transmission)

Channel S4 (CM33: Message transmission/CA55 Core0: Response transmission)

Channel S5 (CM33: Message transmission/CA55 Core1: Response transmission)

Software interrupt control register

Registers that control software interrupts.

The software interrupt control register controls interrupts using the set register, clear register, and status register. An interrupt is asserted by writing “1” to the set register. The interrupt is negated by writing “1” to the clear register. You can also check the interrupt status by reading the status register. With these as one set, the message transmission processing register and response.

One channel is composed of a pair of transmission processing registers, and a total of 4 channels are mounted.

Module stop control

This circuit refers to the module stop control (MHU_MSTOP) in CPG_MHU_MSTOP register on Clock Pulse Generator (CPG) and returns an error response if an access occurs while the module is stopped.

12.2 Interrupt

Table 12.2 shows interrupt output.

Table 12.2 Interrupt type

Output	Description	Pulse/Level	Active Level
msg_ch0_ns	Non secure message transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	HIGH
rsp_ch0_ns	Non secure response transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	HIGH
msg_ch1_ns	Non secure message transmission interrupt (CA55 Core0 -> CM33)	Level	HIGH
rsp_ch1_ns	Non secure response transmission interrupt (CM33 -> CA55 Core0)	Level	HIGH
msg_ch2_ns	Non secure message transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	HIGH
rsp_ch2_ns	Non secure response transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	HIGH
msg_ch3_ns	Non secure message transmission interrupt (CA55 Core1 -> CM33)	Level	HIGH
rsp_ch3_ns	Non secure response transmission interrupt (CM33 -> CA55 Core1)	Level	HIGH
msg_ch4_ns	Non secure message transmission interrupt (CM33 -> CA55 Core0)	Level	HIGH
rsp_ch4_ns	Non secure response transmission interrupt (CA55 Core0 -> CM33)	Level	HIGH
msg_ch5_ns	Non secure message transmission interrupt (CM33 -> CA55 Core1)	Level	HIGH
rsp_ch5_ns	Non secure response transmission interrupt (CA55 Core1 -> CM33)	Level	HIGH
msg_ch0_s	Secure message transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	HIGH
rsp_ch0_s	Secure response transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	HIGH
msg_ch1_s	Secure message transmission interrupt (CA55 Core0 -> CM33)	Level	HIGH
rsp_ch1_s	Secure response transmission interrupt (CM33 -> CA55 Core0)	Level	HIGH
msg_ch2_s	Secure message transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	HIGH
rsp_ch2_s	Secure response transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	HIGH
msg_ch3_s	Secure message transmission interrupt (CA55 Core1 -> CM33)	Level	HIGH
rsp_ch3_s	Secure response transmission interrupt (CM33 -> CA55 Core1)	Level	HIGH
msg_ch4_s	Secure message transmission interrupt (CM33 -> CA55 Core0)	Level	HIGH
rsp_ch4_s	Secure response transmission interrupt (CA55 Core0 -> CM33)	Level	HIGH
msg_ch5_s	Secure message transmission interrupt (CM33 -> CA55 Core1)	Level	HIGH
rsp_ch5_s	Secure response transmission interrupt (CA55 Core1-> CM33)	Level	HIGH
sw_mhu_int_0	software interrupt 0 (CA55-CM33)	Level	HIGH
sw_mhu_int_1	software interrupt 1 (CA55-CM33)	Level	HIGH
sw_mhu_int_2	software interrupt 2 (CA55-CM33)	Level	HIGH
sw_mhu_int_3	software interrupt 3 (CA55-CM33)	Level	HIGH

12.3 Register Configuration

Table 12.3 shows address mapping for MHU. MHU has 4KB address space.

Table 12.3 MHU Address map

Offset Address (CA55 MHU Base address: H’0_1040_0000) (CM33 non-secure MHU Base address: H’4040_0000) (CM33 secure MHU Base address: H’5040_0000)	Description
H’0000 - H’07FF	Non secure message/response interrupt control register
H’0800 - H’0FFF	Non secure software interrupt control register
H’1000 - H’17FF	Secure message/response interrupt control register
H’1800 - H’1FFF	Reserved
H’2000 - H’FFFF	Reserved

Note: Prohibit to access Reserved area.

Table 12.4 shows register attributes

Table 12.4 Register attributes

Attribute	Description
RW	Read&Write register
RW0	Read &Write 0 register. Invalid Write 1.
RW1	Read&Write 1 register. Invalid Write 0.
R0W	Read0&Write register. Always Read 0.
R1W	Read1&Write register. Always Read 1.
R	Read Only
R0W0	Read0&Write0 register. Always Read 0. Invalid Write 1.
R0W1	Read0&Write1 register. Always Read 0. Invalid Write 0.
R1W0	Read1&Write0 register. Always Read 1. Invalid Write 1.
R1W1	Read1&Write1 register. Always Read 1. Invalid Write 0.
RCW0	Read by clear. Write0 register. Invalid Write 1.
RCW1	Read by clear. Write1 register. Invalid Write 0.

Note: No guarantee function for access to Reserved area, Debug and undefined area.

Table 12.5 shows registers

Table 12.5 List of Registers (1/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
Non Secure message/response interrupt				
Non secure message transmission interrupt register (CA55 Core0 -> CA55 Core1) channel 0				
H’0000	message transmission interrupt Status register	MSG_INT_STS0_NS	H’0000_0000	32bit
H’0004	message transmission interrupt Set register	MSG_INT_SET0_NS	H’0000_0000	32bit
H’0008	message transmission interrupt Clear register	MSG_INT_CLR0_NS	H’0000_0000	32bit
H’000C	Reserved	—	—	—
Non secure response transmission interrupt register (CA55 Core1 -> CA55 Core0) channel 0				
H’0010	response transmission interrupt Status register	RSP_INT_STS0_NS	H’0000_0000	32bit
H’0014	response transmission interrupt Set register	RSP_INT_SET0_NS	H’0000_0000	32bit
H’0018	response transmission interrupt Clear register	RSP_INT_CLR0_NS	H’0000_0000	32bit
H’001C	Reserved	—	—	—
Non secure message transmission interrupt register (CA55 Core0 -> CM33) channel 1				
H’0020	message transmission interrupt Status register	MSG_INT_STS1_NS	H’0000_0000	32bit
H’0024	message transmission interrupt Set register	MSG_INT_SET1_NS	H’0000_0000	32bit
H’0028	message transmission interrupt Clear register	MSG_INT_CLR1_NS	H’0000_0000	32bit
H’002C	Reserved	—	—	—
Non secure response transmission interrupt register (CM33 -> CA55 Core0) channel 1				
H’0030	response transmission interrupt Status register	RSP_INT_STS1_NS	H’0000_0000	32bit
H’0034	response transmission interrupt Set register	RSP_INT_SET1_NS	H’0000_0000	32bit
H’0038	response transmission interrupt Clear register	RSP_INT_CLR1_NS	H’0000_0000	32bit
H’003C	Reserved	—	—	—
Non secure message transmission interrupt register (CA55 Core1 -> CA55 Core0) channel 2				
H’0040	message transmission interrupt Status register	MSG_INT_STS2_NS	H’0000_0000	32bit
H’0044	message transmission interrupt Set register	MSG_INT_SET2_NS	H’0000_0000	32bit
H’0048	message transmission interrupt Clear register	MSG_INT_CLR2_NS	H’0000_0000	32bit
H’004C	Reserved	—	—	—
Non secure response transmission interrupt register (CA55 Core0 -> CA55 Core1) channel 2				
H’0050	response transmission interrupt Status register	RSP_INT_STS2_NS	H’0000_0000	32bit
H’0054	response transmission interrupt Set register	RSP_INT_SET2_NS	H’0000_0000	32bit
H’0058	response transmission interrupt Clear register	RSP_INT_CLR2_NS	H’0000_0000	32bit
H’005C	Reserved	—	—	—
Non secure message transmission interrupt register (CA55 Core1 -> CM33) channel 3				
H’0060	message transmission interrupt Status register	MSG_INT_STS3_NS	H’0000_0000	32bit
H’0064	message transmission interrupt Set register	MSG_INT_SET3_NS	H’0000_0000	32bit
H’0068	message transmission interrupt Clear register	MSG_INT_CLR3_NS	H’0000_0000	32bit
H’006C	Reserved	—	—	—
Non secure response transmission interrupt register (CM33 -> CA55 Core1) channel 3				
H’0070	response transmission interrupt Status register	RSP_INT_STS3_NS	H’0000_0000	32bit
H’0074	response transmission interrupt Set register	RSP_INT_SET3_NS	H’0000_0000	32bit
H’0078	response transmission interrupt Clear register	RSP_INT_CLR3_NS	H’0000_0000	32bit
H’007C	Reserved	—	—	—

Table 12.5 List of Registers (2/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
Non secure message transmission interrupt register (CM33 -> CA55 Core0) channel 4				
H’0080	message transmission interrupt Status register	MSG_INT_STS4_NS	H’0000_0000	32bit
H’0084	message transmission interrupt Set register	MSG_INT_SET4_NS	H’0000_0000	32bit
H’0088	message transmission interrupt Clear register	MSG_INT_CLR4_NS	H’0000_0000	32bit
H’008C	Reserved	—	—	—
Non secure response transmission interrupt register (CA55 Core0 -> CM33) channel 4				
H’0090	response transmission interrupt Status register	RSP_INT_STS4_NS	H’0000_0000	32bit
H’0094	response transmission interrupt Set register	RSP_INT_SET4_NS	H’0000_0000	32bit
H’0098	response transmission interrupt Clear register	RSP_INT_CLR4_NS	H’0000_0000	32bit
H’009C	Reserved	—	—	—
Non secure message transmission interrupt register (CM33 -> CA55 Core1) channel 5				
H’00A0	message transmission interrupt Status register	MSG_INT_STS5_NS	H’0000_0000	32bit
H’00A4	message transmission interrupt Set register	MSG_INT_SET5_NS	H’0000_0000	32bit
H’00A8	message transmission interrupt Clear register	MSG_INT_CLR5_NS	H’0000_0000	32bit
H’00AC	Reserved	—	—	—
Non secure response transmission interrupt register (CA55 Core1 -> CM33) channel 5				
H’00B0	response transmission interrupt Status register	RSP_INT_STS5_NS	H’0000_0000	32bit
H’00B4	response transmission interrupt Set register	RSP_INT_SET5_NS	H’0000_0000	32bit
H’00B8	response transmission interrupt Clear register	RSP_INT_CLR5_NS	H’0000_0000	32bit
H’00BC -H’07FF	Reserved	—	—	—
Non secure software interrupt				
Non secure software interrupt register channel 0				
H’0800	software interrupt Status register	SW_INT_STS0_NS	H’0000_0000	32bit
H’0804	software interrupt Set register	SW_INT_SET0_NS	H’0000_0000	32bit
H’0808	software interrupt Clear register	SW_INT_CLR0_NS	H’0000_0000	32bit
H’080C	Reserved	—	—	—
Non secure software interrupt register channel 1				
H’0810	software interrupt Status register	SW_INT_STS1_NS	H’0000_0000	32bit
H’0814	software interrupt Set register	SW_INT_SET1_NS	H’0000_0000	32bit
H’0818	software interrupt Clear register	SW_INT_CLR1_NS	H’0000_0000	32bit
H’0810	Reserved	—	—	—
Non secure software interrupt register channel 2				
H’0820	software interrupt Status register	SW_INT_STS2_NS	H’0000_0000	32bit
H’0824	software interrupt Set register	SW_INT_SET2_NS	H’0000_0000	32bit
H’0828	software interrupt Clear register	SW_INT_CLR2_NS	H’0000_0000	32bit
H’082C	Reserved	—	—	—
Non secure software interrupt register channel 3				
H’0830	software interrupt Status register	SW_INT_STS3_NS	H’0000_0000	32bit
H’0834	software interrupt Set register	SW_INT_SET3_NS	H’0000_0000	32bit
H’0838	software interrupt Clear register	SW_INT_CLR3_NS	H’0000_0000	32bit
H’083C -H’0FFF	Reserved	—	—	—

Table 12.5 List of Registers (3/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
Secure message/response interrupt				
Secure message transmission interrupt register (CA55 Core0 -> CA55 Core1) channel 0				
H’1000	message transmission interrupt Status register	MSG_INT_STS0_S	H’0000_0000	32bit
H’1004	message transmission interrupt Set register	MSG_INT_SET0_S	H’0000_0000	32bit
H’1008	message transmission interrupt Clear register	MSG_INT_CLR0_S	H’0000_0000	32bit
H’100C	Reserved	—	—	—
Secure response transmission interrupt register (CA55 Core1 -> CA55 Core0) channel 0				
H’1010	response transmission interrupt Status register	RSP_INT_STS0_S	H’0000_0000	32bit
H’1014	response transmission interrupt Set register	RSP_INT_SET0_S	H’0000_0000	32bit
H’1018	response transmission interrupt Clear register	RSP_INT_CLR0_S	H’0000_0000	32bit
H’101C	Reserved	—	—	—
Secure message transmission interrupt register (CA55 Core0 -> CM33) channel 1				
H’1020	message transmission interrupt Status register	MSG_INT_STS1_S	H’0000_0000	32bit
H’1024	message transmission interrupt Set register	MSG_INT_SET1_S	H’0000_0000	32bit
H’1028	message transmission interrupt Clear register	MSG_INT_CLR1_S	H’0000_0000	32bit
H’102C	Reserved	—	—	—
Secure response transmission interrupt register (CM33 -> CA55 Core0) channel 1				
H’1030	response transmission interrupt Status register	RSP_INT_STS1_S	H’0000_0000	32bit
H’1034	response transmission interrupt Set register	RSP_INT_SET1_S	H’0000_0000	32bit
H’1038	response transmission interrupt Clear register	RSP_INT_CLR1_S	H’0000_0000	32bit
H’103C	Reserved	—	—	—
Secure message transmission interrupt register (CA55 Core1 -> CA55 Core0) channel 2				
H’1040	message transmission interrupt Status register	MSG_INT_STS2_S	H’0000_0000	32bit
H’1044	message transmission interrupt Set register	MSG_INT_SET2_S	H’0000_0000	32bit
H’1048	message transmission interrupt Clear register	MSG_INT_CLR2_S	H’0000_0000	32bit
H’104C	Reserved	—	—	—
Secure response transmission interrupt register (CA55 Core0-> CA55 Core1) channel 2				
H’1050	response transmission interrupt Status register	RSP_INT_STS2_S	H’0000_0000	32bit
H’1054	response transmission interrupt Set register	RSP_INT_SET2_S	H’0000_0000	32bit
H’1058	response transmission interrupt Clear register	RSP_INT_CLR2_S	H’0000_0000	32bit
H’105C	Reserved	—	—	—
Secure message transmission interrupt register (CA55 Core1 -> CM33) channel 3				
H’1060	message transmission interrupt Status register	MSG_INT_STS3_S	H’0000_0000	32bit
H’1064	message transmission interrupt Set register	MSG_INT_SET3_S	H’0000_0000	32bit
H’1068	message transmission interrupt Clear register	MSG_INT_CLR3_S	H’0000_0000	32bit
H’106C	Reserved	—	—	—
Secure response transmission interrupt register (CM33 -> CA55 Core1) channel 3				
H’1070	response transmission interrupt Status register	RSP_INT_STS3_S	H’0000_0000	32bit
H’1074	response transmission interrupt Set register	RSP_INT_SET3_S	H’0000_0000	32bit
H’1078	response transmission interrupt Clear register	RSP_INT_CLR3_S	H’0000_0000	32bit
H’107C	Reserved	—	—	—

Table 12.5 List of Registers (4/4)

Offset Address	Register Name	Abbreviation	Initial value	Access Size
Secure message transmission interrupt register (CM33 -> CA55 Core0) channel 4				
H’1080	message transmission interrupt Status register	MSG_INT_STS4_S	H’0000_0000	32bit
H’1084	message transmission interrupt Set register	MSG_INT_SET4_S	H’0000_0000	32bit
H’1088	message transmission interrupt Clear register	MSG_INT_CLR4_S	H’0000_0000	32bit
H’108C	Reserved	—	—	—
Secure response transmission interrupt register (CA55 Core0 -> CM33) channel 4				
H’1090	response transmission interrupt Status register	RSP_INT_STS4_S	H’0000_0000	32bit
H’1094	response transmission interrupt Set register	RSP_INT_SET4_S	H’0000_0000	32bit
H’1098	response transmission interrupt Clear register	RSP_INT_CLR4_S	H’0000_0000	32bit
H’109C	Reserved	—	—	—
Secure message transmission interrupt register (CM33 -> CA55 Core1) channel 5				
H’10A0	message transmission interrupt Status register	MSG_INT_STS5_S	H’0000_0000	32bit
H’10A4	message transmission interrupt Set register	MSG_INT_SET5_S	H’0000_0000	32bit
H’10A8	message transmission interrupt Clear register	MSG_INT_CLR5_S	H’0000_0000	32bit
H’10AC	Reserved	—	—	—
Secure response transmission interrupt register (CA55 Core1 -> CM33) channel 5				
H’10B0	response transmission interrupt Status register	RSP_INT_STS5_S	H’0000_0000	32bit
H’10B4	response transmission interrupt Set register	RSP_INT_SET5_S	H’0000_0000	32bit
H’10B8	response transmission interrupt Clear register	RSP_INT_CLR5_S	H’0000_0000	32bit
H’10BC -H’17FF	Reserved	—	—	—

Note: Access size must be 32bit. Prohibit to access to Reserved register.

12.4 Register Descriptions

12.4.1 Non secure message transmission interrupt Status register (CA55 Core0 -> CA55 Core1)

12.4.2 Non secure message transmission interrupt Status register (CA55 Core0 -> CM33)

12.4.3 Non secure message transmission interrupt Status register (CA55 Core1 -> CA55 Core0)

12.4.4 Non secure message transmission interrupt Status register (CA55 Core1 -> CM33)

12.4.5 Non secure message transmission interrupt Status register (CM33 -> CA55 Core0)

12.4.6 Non secure message transmission interrupt Status register (CA55 Core1 -> CM33)

Status register for Non secure message transmission interrupt (Name: MSG_INT_STSn_NS <n=0-5>)

Access Size:	32bit
Address(es):	MSG_INT_STS0_NS : H'000 MSG_INT_STS1_NS : H'020 MSG_INT_STS2_NS : H'040 MSG_INT_STS3_NS : H'060 MSG_INT_STS4_NS : H'080 MSG_INT_STS5_NS : H'0A0
Initial Value:	H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of message transmission interrupt. Read only. Write is invalid. 0: No message transmission interrupt (No interrupt being notified to the interrupt controller at the message destination) 1: Assert message transmission interrupt (There is an interrupt being notified to the interrupt controller at the message destination.)

12.4.7 Non secure message transmission interrupt Set register (CA55 Core0 -> CA55 Core1)

12.4.8 Non secure message transmission interrupt Set register (CA55 Core0 -> CM33)

12.4.9 Non secure message transmission interrupt Set register (CA55 Core1 -> CA55 Core0)

12.4.10 Non secure message transmission interrupt Set register (CA55 Core1 -> CM33)

12.4.11 Non secure message transmission interrupt Set register (CM33 -> CA55 Core0)

12.4.12 Non secure message transmission interrupt Set register (CA55 Core1 -> CM33)

Set register for Non secure message transmission interrupt (Name: MSG_INT_SETn_NS <n=0-5>)

Access Size: 32bit

Address(es):
MSG_INT_SET0_NS : H'004
MSG_INT_SET1_NS : H'024
MSG_INT_SET2_NS : H'044
MSG_INT_SET3_NS : H'064
MSG_INT_SET4_NS : H'084
MSG_INT_SET5_NS : H'0A4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set message transmission interrupt. Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.13 Non secure message transmission interrupt Clear register (CA55 Core0 -> CA55 Core1)

12.4.14 Non secure message transmission interrupt Clear register (CA55 Core0 -> CM33)

12.4.15 Non secure message transmission interrupt Clear register (CA55 Core1 -> CA55 Core0)

12.4.16 Non secure message transmission interrupt Clear register (CA55 Core1 -> CM33)

12.4.17 Non secure message transmission interrupt Clear register (CM33 -> CA55 Core0)

12.4.18 Non secure message transmission interrupt Clear register (CM33 -> CA55 Core1)

Clear register of Non secure message transmission interrupt (Name: MSG_INT_CLRn_NS <n=0-5>)

Access Size:	32bit
Address(es):	MSG_INT_CLR0_NS : H'008 MSG_INT_CLR1_NS : H'028 MSG_INT_CLR2_NS : H'048 MSG_INT_CLR3_NS : H'068 MSG_INT_CLR4_NS : H'088 MSG_INT_CLR5_NS : H'0A8
Initial Value:	H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear message transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.4.19 Non secure response transmission interrupt Status register (CA55 Core1 -> CA55 Core0)

12.4.20 Non secure response transmission interrupt Status register (CM33 -> CA55 Core0)

12.4.21 Non secure response transmission interrupt Status register (CA55 Core0 -> CA55 Core1)

12.4.22 Non secure response transmission interrupt Status register (CM33 -> CA55 Core1)

12.4.23 Non secure response transmission interrupt Status register (CA55 Core0 -> CM33)

12.4.24 Non secure response transmission interrupt Status register (CA55 Core1 -> CM33)

Status register for Non secure response transmission interrupt (Name: RSP_INT_STSn_NS <n=0-5>)

Access Size:	32bit
Address(es):	RSP_INT_STS0_NS : H’010 RSP_INT_STS1_NS : H’030 RSP_INT_STS2_NS : H’050 RSP_INT_STS3_NS : H’070 RSP_INT_STS4_NS : H’090 RSP_INT_STS5_NS : H’0B0
Initial Value:	H’0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of response transmission interrupt. Read only. Write is invalid. 0: No response transmission interrupt (No interrupt being notified to the interrupt controller at the message destination) 1: Assert response transmission interrupt (There is an interrupt being notified to the interrupt controller at the message destination.)

12.4.25 Non secure response transmission interrupt Set register (CA55 Core1 -> CA55 Core0)

12.4.26 Non secure response transmission interrupt Set register (CM33 -> CA55 Core0

12.4.27 Non secure response transmission interrupt Set register (CA55 Core0 -> CA55 Core1)

12.4.28 Non secure response transmission interrupt Set register (CM33 -> CA55 Core1)

12.4.29 Non secure response transmission interrupt Set register (CA55 Core0 -> CM33)

12.4.30 Non secure response transmission interrupt Set register (CA55 Core1 -> CM33)

Set register of Non secure response transmission interrupt (Name: RSP_INT_SETn_NS <n=0-5>)

Access Size: 32bit

Address(es): RSP_INT_SET0_NS : H'014
RSP_INT_SET1_NS : H'034
RSP_INT_SET2_NS : H'054
RSP_INT_SET3_NS : H'074
RSP_INT_SET4_NS : H'094
RSP_INT_SET5_NS : H'0B4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set response transmission interrupt. Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.31 Non secure response transmission interrupt Clear register (CA55 Core1 -> CA55 Core0)

12.4.32 Non secure response transmission interrupt Clear register (CM33 -> CA55 Core0)

12.4.33 Non secure response transmission interrupt Clear register (CA55 Core0 -> CA55 Core1)

12.4.34 Non secure response transmission interrupt Clear register (CM33 -> CA55 Core1)

12.4.35 Non secure response transmission interrupt Clear register (CA55 Core0 -> CM33)

12.4.36 Non secure response transmission interrupt Clear register (CA55 Core1 -> CM33)

Clear register of Non secure response transmission interrupt (Name: RSP_INT_CLRn_NS <n=0-5>)

Access Size:	32bit
Address(es):	RSP_INT_CLR0_NS : H’018 RSP_INT_CLR1_NS : H’038 RSP_INT_CLR2_NS : H’058 RSP_INT_CLR3_NS : H’078 RSP_INT_CLR4_NS : H’098 RSP_INT_CLR5_NS : H’0B8
Initial Value:	H’0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear response transmission interrupt Write “1” only. Always read “0”. 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.4.37 Non secure software interrupt Status register (ch0)

12.4.38 Non secure software interrupt Status register (ch1)

12.4.39 Non secure software interrupt Status register (ch2)

12.4.40 Non secure software interrupt Status register (ch3)

Status register of Non secure software interrupt Status register (Name: SW_INT_STSn_NS <n=0-3>)

Access Size: 32bit

Address(es): SW_INT_STS0_NS : H'0800
SW_INT_STS1_NS : H'0810
SW_INT_STS2_NS : H'0820
SW_INT_STS3_NS : H'0830

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of Non secure software interrupt. Write is invalid. 0: No interrupt (No interrupt being notified to the interrupt controller) 1: Software interrupt is asserted. (There is an interrupt being notified to the interrupt controller)

12.4.41 Non secure software interrupt Set register (ch0)

12.4.42 Non secure software interrupt Set register (ch1)

12.4.43 Non secure software interrupt Set register (ch2)

12.4.44 Non secure software interrupt Set register (ch3)

Set register of Non secure software interrupt (Name: SW_INT_SETn_NS <n=0-3>)

Access Size: 32bit

Address(es): SW_INT_SET0_NS : H'0804
SW_INT_SET1_NS : H'0814
SW_INT_SET2_NS : H'0824
SW_INT_SET3_NS : H'0834

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set Non secure software interrupt. Write “1” only. Always read “0”. 0: invalid 1:Issue interrupt to interrupt controller

12.4.45 Non secure software interrupt Clear register (ch0)

12.4.46 Non secure software interrupt Clear register (ch1)

12.4.47 Non secure software interrupt Clear register (ch2)

12.4.48 Non secure software interrupt Clear register (ch3)

Clear register of Non secure software interrupt Clear register(Name: SW_INT_CLRn_NS <n=0-3>)

Access Size: 32bit

Address(es): SW_INT_CLR0_NS : H’0808
SW_INT_CLR1_NS : H’0818
SW_INT_CLR2_NS : H’0828
SW_INT_CLR3_NS : H’0838

Initial Value: H’0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear Non secure software interrupt. Write “1” only. Always read “0”. 0: invalid 1:Clear interrupt to interrupt controller.

12.4.49 Secure message transmission interrupt Status register (CA55 Core0 -> CA55 Core1)

12.4.50 Secure message transmission interrupt Status register (CA55 Core0 -> CM33)

12.4.51 Secure message transmission interrupt Status register (CA55 Core1 -> CA55 Core0)

12.4.52 Secure message transmission interrupt Status register (CA55 Core1 -> CM33)

12.4.53 Secure message transmission interrupt Status register (CM33 -> CA55 Core0)

12.4.54 Secure message transmission interrupt Status register (CM33 -> CA55 Core1)

Status register of Secure message transmission interrupt (Name: MSG_INT_STSn_S <n=0-5>)

Access Size:	32bit
Address(es):	MSG_INT_STS0_S : H'1000 MSG_INT_STS1_S : H'1020 MSG_INT_STS2_S : H'1040 MSG_INT_STS3_S : H'1060 MSG_INT_STS4_S : H'1080 MSG_INT_STS5_S : H'10A0
Initial Value:	H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of message transmission interrupt. Write is invalid. 0: No Message transmission interrupt (No interrupt being notified to the interrupt controller of the message destination) 1: message transmission interrupt is asserted. (There is an interrupt being notified to the interrupt controller of the message destination)

12.4.55 Secure message transmission interrupt Set register (CA55 Core0 -> CA55 Core1)

12.4.56 Secure message transmission interrupt Set register (CA55 Core0 -> CM33)

12.4.57 Secure message transmission interrupt Set register (CA55 Core1 -> CA55 Core0)

12.4.58 Secure message transmission interrupt Set register (CA55 Core1 -> CM33)

12.4.59 Secure message transmission interrupt Set register (CM33 -> CA55 Core0)

12.4.60 Secure message transmission interrupt Set register (CM33 -> CA55 Core1)

Set register of Secure message transmission interrupt (Name: MSG_INT_SETn_S <n=0-5>)

Access Size:	32bit
Address(es):	MSG_INT_SET0_S : H'1004 MSG_INT_SET1_S : H'1024 MSG_INT_SET2_S : H'1044 MSG_INT_SET3_S : H'1064 MSG_INT_SET4_S : H'1084 MSG_INT_SET5_S : H'10A4
Initial Value:	H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set message transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.61 Secure message transmission interrupt Clear register (CA55 Core0 -> CA55 Core1)

12.4.62 Secure message transmission interrupt Clear register (CA55 Core0 -> CM33)

12.4.63 Secure message transmission interrupt Clear register (CA55 Core1 -> CA55 Core0)

12.4.64 Secure message transmission interrupt Clear register (CA55 Core1 -> CM33)

12.4.65 Secure message transmission interrupt Clear register (CM33 -> CA55 Core0)

12.4.66 Secure message transmission interrupt Clear register (CM33 -> CA55 Core1)

Clear register of Secure message transmission interrupt (Name: MSG_INT_CLRn_S <n=0-5>)

Access Size:	32bit
Address(es):	MSG_INT_CLR0_S : H'1008 MSG_INT_CLR1_S : H'1028 MSG_INT_CLR2_S : H'1048 MSG_INT_CLR3_S : H'1068 MSG_INT_CLR4_S : H'1088 MSG_INT_CLR5_S : H'10A8
Initial Value:	H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear message transmission interrupt. Write "1" only. Always read "0". 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.4.67 Secure response transmission interrupt Status register (CA55 Core1 -> CA55 Core0)

12.4.68 Secure response transmission interrupt Status register (CM33 -> CA55 Core0)

12.4.69 Secure response transmission interrupt Status register (CA55 Core0 -> CA55 Core1)

12.4.70 Secure response transmission interrupt Status register (CM33 -> CA55 Core1)

12.4.71 Secure response transmission interrupt Status register (CA55 Core0 -> CM33)

12.4.72 Secure response transmission interrupt Status register (CA55 Core1 -> CM33)

Status register of Secure response transmission interrupt (Name: RSP_INT_STSn_S <n=0-5>)

Access Size: 32bit

Address(es):
RSP_INT_STS0_S : H'1010
RSP_INT_STS1_S : H'1030
RSP_INT_STS2_S : H'1050
RSP_INT_STS3_S : H'1070
RSP_INT_STS4_S : H'1090
RSP_INT_STS5_S : H'10B0

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STAT	0	R	Show status of response transmission interrupt. Write is invalid. 0: No response transmission interrupt (No interrupt being notified to the interrupt controller of the message destination.) 1: response transmission interrupt is asserted. (There is an interrupt being notified to the interrupt controller of the message destination.)

12.4.73 Secure response transmission interrupt Set register (CA55 Core1 -> CA55 Core0)

12.4.74 Secure response transmission interrupt Set register (CM33 -> CA55 Core0)

12.4.75 Secure response transmission interrupt Set register (CA55 Core0 -> CA55 Core1

12.4.76 Secure response transmission interrupt Set register (CM33 -> CA55 Core1)

12.4.77 Secure response transmission interrupt Set register (CA55 Core0 -> CM33)

12.4.78 Secure response transmission interrupt Set register (CA55 Core1 -> CM33)

Set register of Secure response transmission interrupt (Name: RSP_INT_SETn_NS <n=0-5>)

Access Size: 32bit

Address(es): RSP_INT_SET0_S : H'1014
RSP_INT_SET1_S : H'1034
RSP_INT_SET2_S : H'1054
RSP_INT_SET3_S : H'1074
RSP_INT_SET4_S : H'1094
RSP_INT_SET5_S : H'10B4

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SET	0	R0W1	Set response transmission interrupt Write "1" only. Always read "0". 0: invalid 1: Issue interrupt to interrupt controller of the message destination.

12.4.79 Secure response transmission interrupt Clear register (CA55 Core1 -> CA55 Core0)

12.4.80 Secure response transmission interrupt Clear register (CM33 -> CA55 Core0)

12.4.81 Secure response transmission interrupt Clear register (CA55 Core0 -> CA55 Core1)

12.4.82 Secure response transmission interrupt Clear register (CM33 -> CA55 Core1)

12.4.83 Secure response transmission interrupt Clear register (CA55 Core0 -> CM33)

12.4.84 Secure response transmission interrupt Clear register (CA55 Core1 -> CM33)

Clear register of Secure response transmission interrupt Clear register(Name: RSP_INT_CLRn_S <n=0-5>)

Access Size: 32bit

Address(es): RSP_INT_CLR0_S : H'1018
RSP_INT_CLR1_S : H'1038
RSP_INT_CLR2_S : H'1058
RSP_INT_CLR3_S : H'1078
RSP_INT_CLR4_S : H'1098
RSP_INT_CLR5_S : H'10B8

Initial Value: H'0000_0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CLEAR	0	R0W1	Clear response transmission interrupt Write “1” only. Always read “0”. 0: invalid 1: Clear interrupt to interrupt controller of the message destination.

12.5 Function description

12.5.1 Inter CPU communication

The hardware configuration is shown in **Figure 12.2**. Inter-CPU communication is established by sending a message from the message source to the message destination and sending a response from the message destination to the message source. The MHU uses an interrupt to notify that the message / response to the CPU has been written to Shared RAM(On-chip RAM). Interrupts are controlled by the Set register, Clear register, and Status register. Interrupt can be asserted by writing "1" to the Set register. Interrupt is negated by writing "1" to the Clear register, and check the interrupt status by reading the Status register. With these registers as one set, one channel is composed of a message transmission processing / response transmission processing pair. A total of 12 channels are installed so that bidirectional communication can be performed between Cortex-A55 Core0, Cortex-A55 Core1 and Cortex-M33. **Table 12.6** shows the channel and message source / destination correspondence.

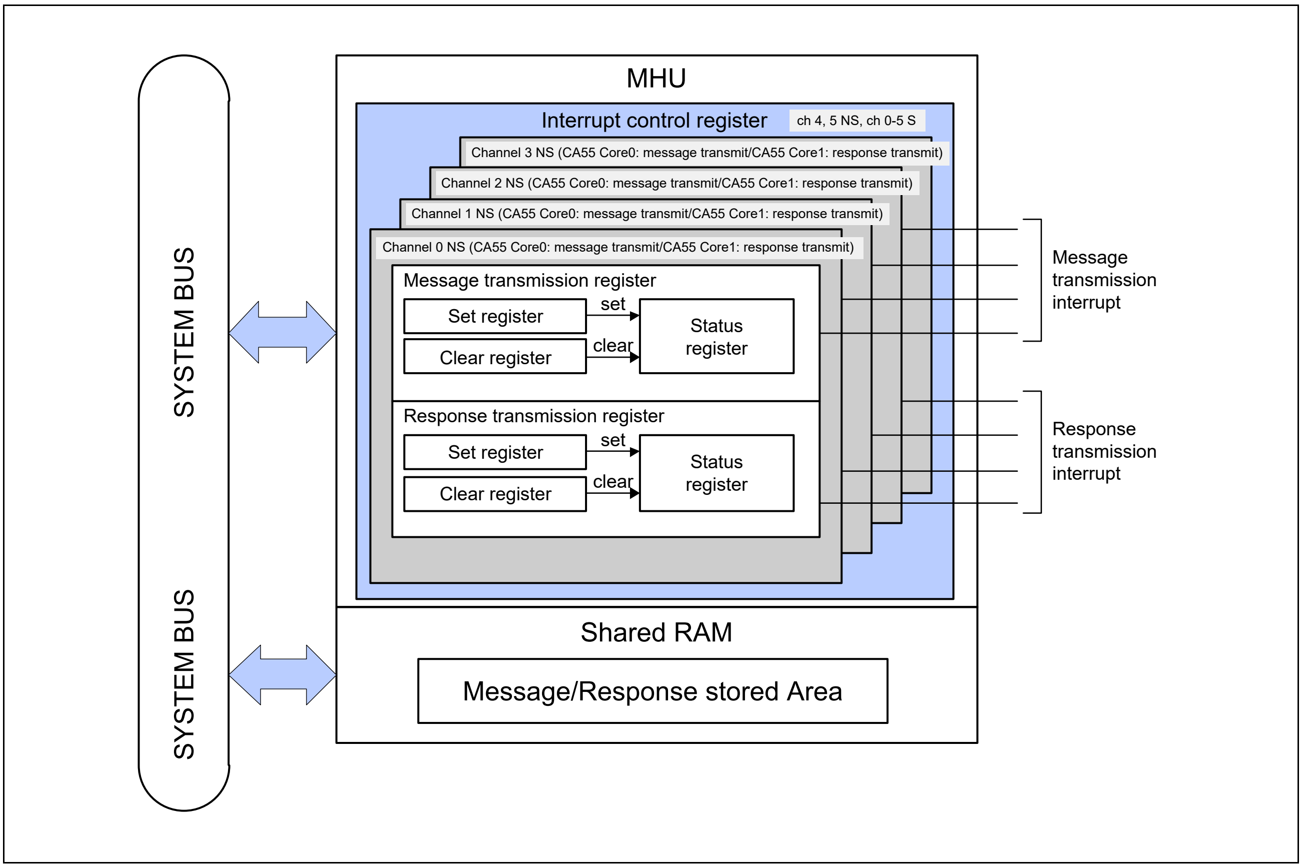


Figure 12.2 Hardware configuration

Table 12.6 Channel and message destination/source reference

channel	Source	Destination	Message transmit register	Response transmit register
channel 0 NS	CA55 Core0	CA55 Core1	MSG_INT_(SET/CLR/STS)0_NS	RSP_INT_(SET/CLR/STS)0_NS
channel 1 NS	CA55 Core0	CM33	MSG_INT_(SET/CLR/STS)1_NS	RSP_INT_(SET/CLR/STS)1_NS
channel 2 NS	CA55 Core1	CA55 Core0	MSG_INT_(SET/CLR/STS)2_NS	RSP_INT_(SET/CLR/STS)2_NS
channel 3 NS	CA55 Core1	CM33	MSG_INT_(SET/CLR/STS)3_NS	RSP_INT_(SET/CLR/STS)3_NS
channel 4 NS	CM33	CA55 Core0	MSG_INT_(SET/CLR/STS)4_NS	RSP_INT_(SET/CLR/STS)4_NS
channel 5 NS	CM33	CA55 Core1	MSG_INT_(SET/CLR/STS)5_NS	RSP_INT_(SET/CLR/STS)5_NS
channel 0 S	CA55 Core0	CA55 Core1	MSG_INT_(SET/CLR/STS)0_S	RSP_INT_(SET/CLR/STS)0_S
channel 1 S	CA55 Core0	CM33	MSG_INT_(SET/CLR/STS)1_S	RSP_INT_(SET/CLR/STS)1_S
channel 2 S	CA55 Core1	CA55 Core0	MSG_INT_(SET/CLR/STS)2_S	RSP_INT_(SET/CLR/STS)2_S
channel 3 S	CA55 Core1	CM33	MSG_INT_(SET/CLR/STS)3_S	RSP_INT_(SET/CLR/STS)3_S
channel 4 S	CM33	CA55 Core0	MSG_INT_(SET/CLR/STS)4_S	RSP_INT_(SET/CLR/STS)4_S
channel 5 S	CM33	CA55 Core1	MSG_INT_(SET/CLR/STS)5_S	RSP_INT_(SET/CLR/STS)5_S

12.6 Operation sequence

12.6.1 Message transmit sequence

Show Message transmit sequence from Cortex-M33 to Cortex-A55 Core0 in below and **Figure 12.3**.

(1)	Cortex-M33:	Check “0” for bit0 in register: MSG_INT_STS4. (Confirm cleared status and move to (2))
(2)	Cortex-M33:	Write message to Shared RAM
(3)	Cortex-M33:	Write “1” to bit 0 in register: MSG_INT_SET4
(4)	MHU channel 4:	Assert interrupt: int_msg_ch4 for Cortex-A55
(5)	Cortex-A55 Core0:	Check interrupt reason (Detect message transmit)
(6)	Cortex-A55 Core0:	Check “1” for bit 0 in register: MSG_INT_STS4
(7)	Cortex-A55 Core0:	Read message from shared RAM
(8)	Cortex-A55 Core0:	Write “1” to bit0 in register: MSG_INT_CLR4
(9)	MHU channel 4:	Negate interrupt: int_msg_ch4 to Cortex-A55

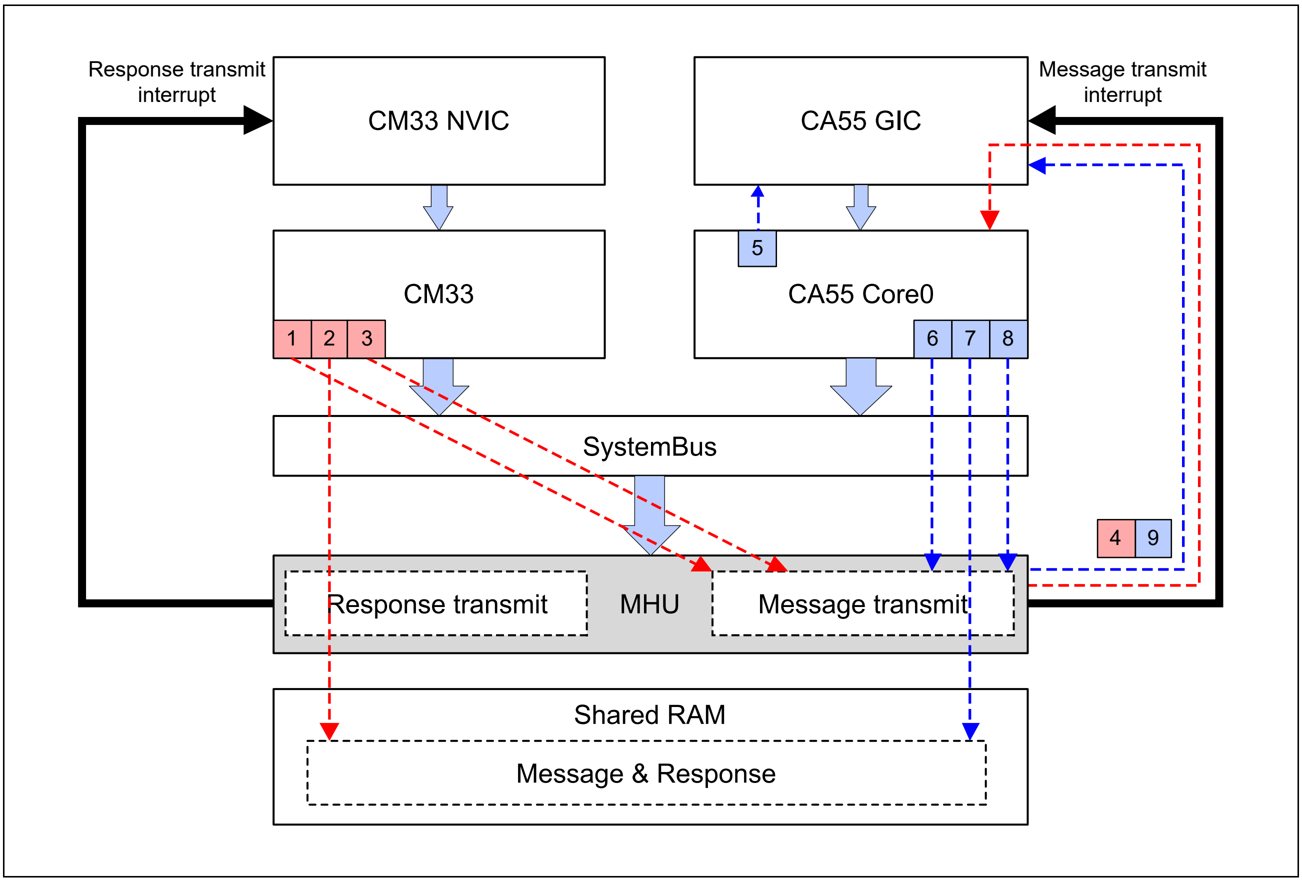


Figure 12.3 Message transmit sequence (CM33 -> CA55 Core0)

12.6.2 Response transmit sequence

Show Response transmit sequence from Cortex-A55 Core0 to Cortex-M33 in below and **Figure 12.4**.

(1) Cortex-A55 Core0: Check “0” for bit0 in register: RSP_INT_STS4.
(Confirm cleared status and move to (2))

(2) Cortex-A55 Core0: Write Response to Shared RAM

(3) Cortex-A55 Core0: Write “1” to bit 0 in register: RSP_INT_SET4

(4) MHU channel 4: Assert interrupt: int_rsp_ch2 for Cortex-M33

(5) Cortex-M33: Check interrupt reason (Detect response transmit)

(6) Cortex-M33: Check “1” for bit 0 in register: RSP_INT_STS4

(7) Cortex-M33: Read response from shared RAM

(8) Cortex-M33: Write “1” to bit0 in register: RSP_INT_CLR4

(9) MHU channel 4: Negate interrupt:int_rsp_ch4 to Cortex-M33

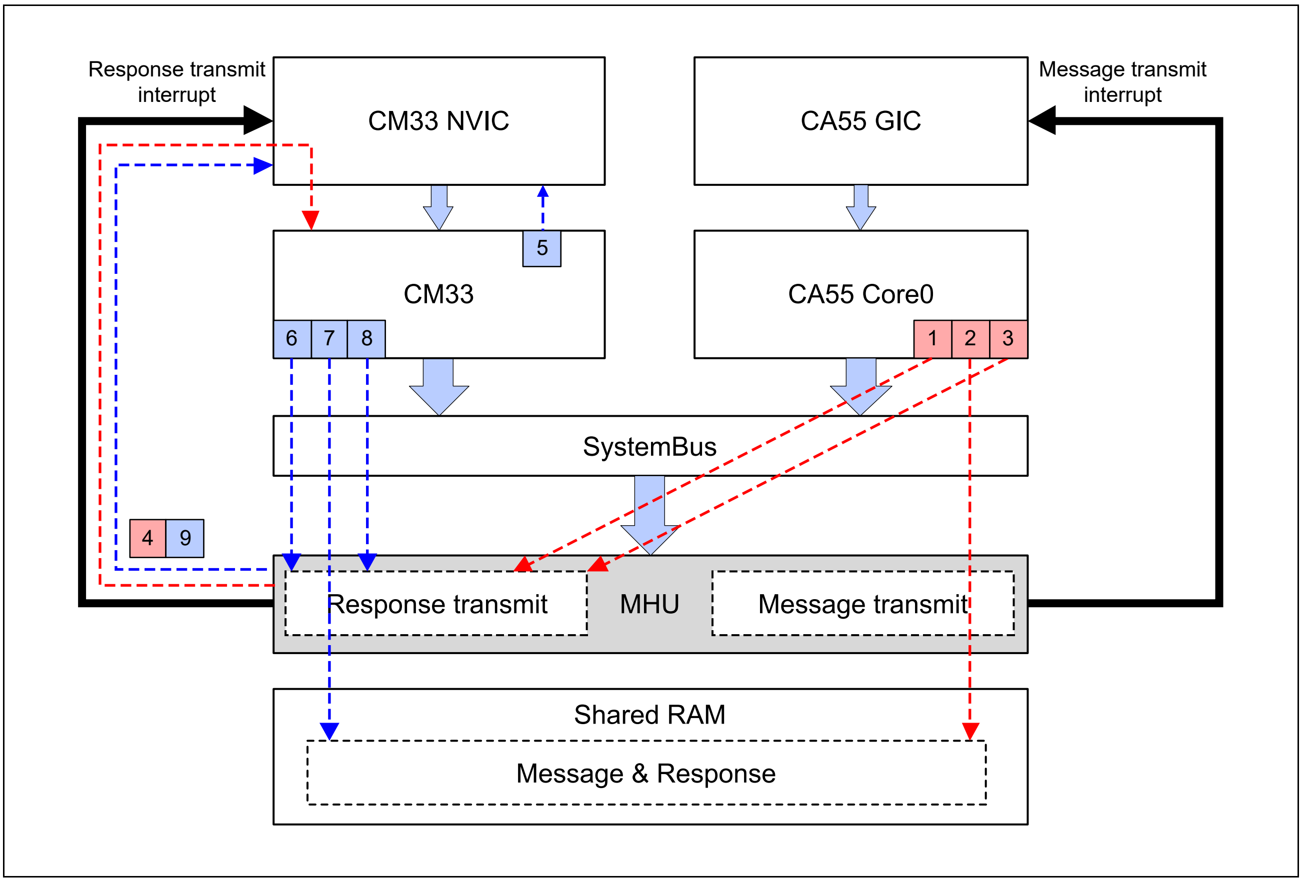


Figure 12.4 Response transmit sequence (CA55 Core0 -> CM33)

13. TrustZone Address Space Controller (TZC)

13.1 Overview

This LSI has four TrustZone Address Space Controller (TZC) to realize memory access in a safe area. It performs security checks on transactions to memory or peripherals. Transactions must meet security requirements to access memory or peripherals. This is an IP (“CoreLink™ TrustZone Address Space Controller TZC-400”) provided by Arm, for details on the functions of TZC-400, see the relevant Technical Reference Manual.

13.1.1 Features

The TZC works between TrustZone system ACE-Lite masters and ACE-Lite slaves to filter bus access from master to slave. It performs filtering based on the security requirements specified in the address space.

Each TZC is installed on the interface of the following modules of this LSI.

- Internal RAM for ACPU (64 KB)
- Internal RAM for MCPU (64 KB)
- SPI Multi I/O Bus Controller
- DDR3L/DDR4 SDRAM Memory Controller

For details on the internal bus, please refer to **Section 5, LSI Internal Bus**.

The TZC provides the following key features:

- The ability to define up to eight address regions in the area map.
- A default base region to cover all remaining portions of the address map.
- Software programmable security access permissions for each address region through an APB interface.
  This includes the default base region, Region 0.
- Filter units only allow data transfer between an ACE-Lite master and an ACE-Lite slave if the security status of the ACE-Lite transaction and its identity match the security settings of the memory region it addresses.
- Common region configuration register settings shared between multiple filter units.
- Normal path for accesses with a much higher outstanding access support.
- Identity-based filtering of Non-secure accesses.
- Reporting and interrupt signaling that is configurable from software to manage failed permission checks.
  You can program the TZC to assert TZCnINT (n = 0, 1, 2, 3) when an access fails its security check.
- Gate keeper to allow or block accesses to each filter unit.
- Support for 256 outstanding transactions on the normal paths.

13.1.2 Block Diagram

Figure 13.1 shows a block diagram.

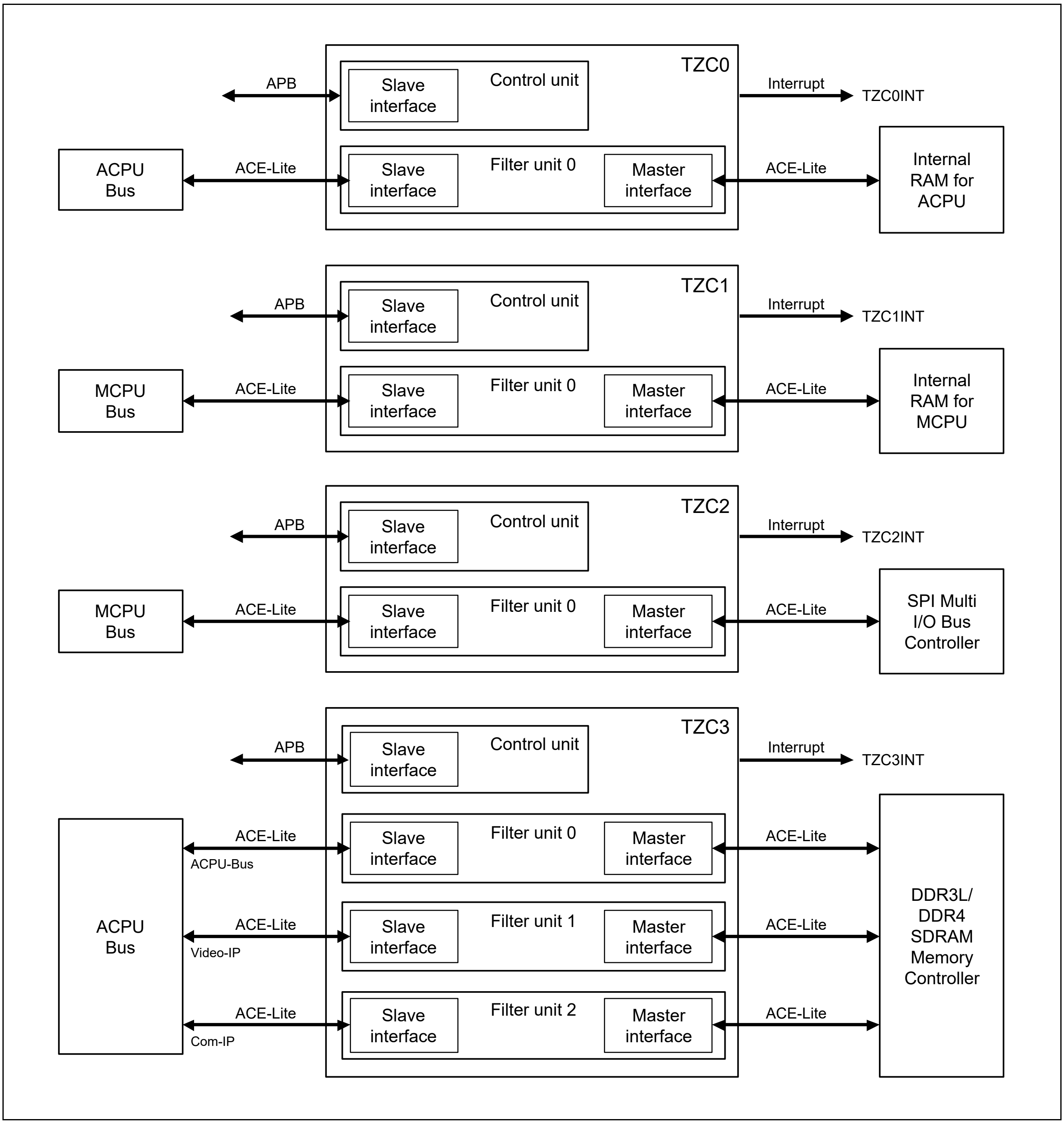


Figure 13.1 Block Diagram

Filter units perform security checks. Each filter unit has an ACE-Lite slave interface and an ACE-Lite master interface. All filter units operate from one set of shared region configuration registers. This ensures consistency across all filter units.

The number of filter units of TZC installed on each interface is as follows.

- Internal RAM for ACPU (64 KB): 1 filter unit
- Internal RAM for MCPU (64 KB): 1 filter unit
- SPI Multi I/O Bus Controller: 1 filter unit
- DDR3L/DDR4 SDRAM Memory Controller: 3 filter units
    - filter unit 0: Access from the ACPU bus passes
    - filter unit 1: Access from the Video IP passes
    - filter unit 2: Access from the Communication IP passes

13.1.3 External Pins

In TZC, there are no external pins.

13.2 Register Configuration

The base address of the TZC associated with each module is as follows:

- TZC0: Internal RAM for ACPU (64KB)

  H’0_1104_0000 (Cortex-A55 Address Space)
  H’4104_0000 (Cortex-M33 Address Space Non-Secure)
  H’5104_0000 (Cortex-M33 Address Space Secure)

- TZC1: Internal RAM for MCPU (64KB)

  H’0_1105_0000 (Cortex-A55 Address Space)
  H’4105_0000 (Cortex-M33 Address Space Non-Secure)
  H’5105_0000 (Cortex-M33 Address Space Secure)

- TZC2: SPI Multi I/O Bus Controller

  H’0_1106_0000 (Cortex-A55 Address Space)
  H’4106_0000 (Cortex-M33 Address Space Non-Secure)
  H’5106_0000 (Cortex-M33 Address Space Secure)

- TZC3: DDR3L/DDR4 SDRAM Memory Controller

  H’0_1107_0000 (Cortex-A55 Address Space)
  H’4107_0000 (Cortex-M33 Address Space Non-Secure)
  H’5107_0000 (Cortex-M33 Address Space Secure)

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

13.3 Register Descriptions

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

TZC has registers to specify the address area where access is controlled. Please refer Overall Address Space (**Table 5.1, Detailed Address Space**) for the addresses specified in these registers.

13.4 Operation

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

13.5 Usage Note

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

14. Direct Memory Access Controller

The direct memory access controller can be used in place of the CPU to perform high-speed transfers between external memory, on-chip memory, memory-mapped external devices, and on-chip peripheral modules.

This module has controllers that handles secure and non-secure access. Do not assign the same DMA transfer request to secure access and non- secure access, respectively.

14.1 Features

- Number of channels selectable: 16 channels (CH0 to CH15).
- 4-Gbyte address space (according to the architecture)
- Transfer data size: Byte, 2 bytes, 4 bytes, 8 bytes, 16 bytes, 32 bytes, 64 bytes, and 128 bytes
- Maximum transfer count: $2^{32} - 1$ bytes
- Address mode: Dual address mode
- Transfer requests: Can be selected from the two types of on-chip peripheral module request, and auto request (software trigger)
- Transfer mode: Single transfer mode and block transfer mode are selectable.
- Priority: The channel priority levels within channels 0 to 7 and within channels 8 to 15 are selectable between fixed mode and round-robin mode (the channel priority level between the group of channels 0 to 7 and the group of channels 8 to 15 is round-robin mode).
- Interrupt request: An interrupt request can be sent to the CPU on completion of data transfer (DMA transfer end interrupt per channel) or on occurrence of a transfer error (DMA error interrupt).
- The DMA registers have a continuous execution function that allows the next DMA transfer to be executed continuously by making settings for the next DMA transfer during execution of the current DMA transfer. This continuous execution function can be enabled or disabled independently in each channel.
- Link mode: In this mode, the setting data (descriptor data) located in the memory by the CPU is automatically retrieved by the DMAC, and DMA transfer is performed according to those values.
- Buffer sweep: If an ongoing DMA transfer is forced to end, the data already retrieved into the buffer can be output before DMA transfer ends.
- Interval: A specific DMA transfer interval can be specified to adjust the bus occupancy.

14.2 Input/Output Pins

Requests from external pins are not supported.

14.3 Register Configuration

The register configuration is shown in the figure below.

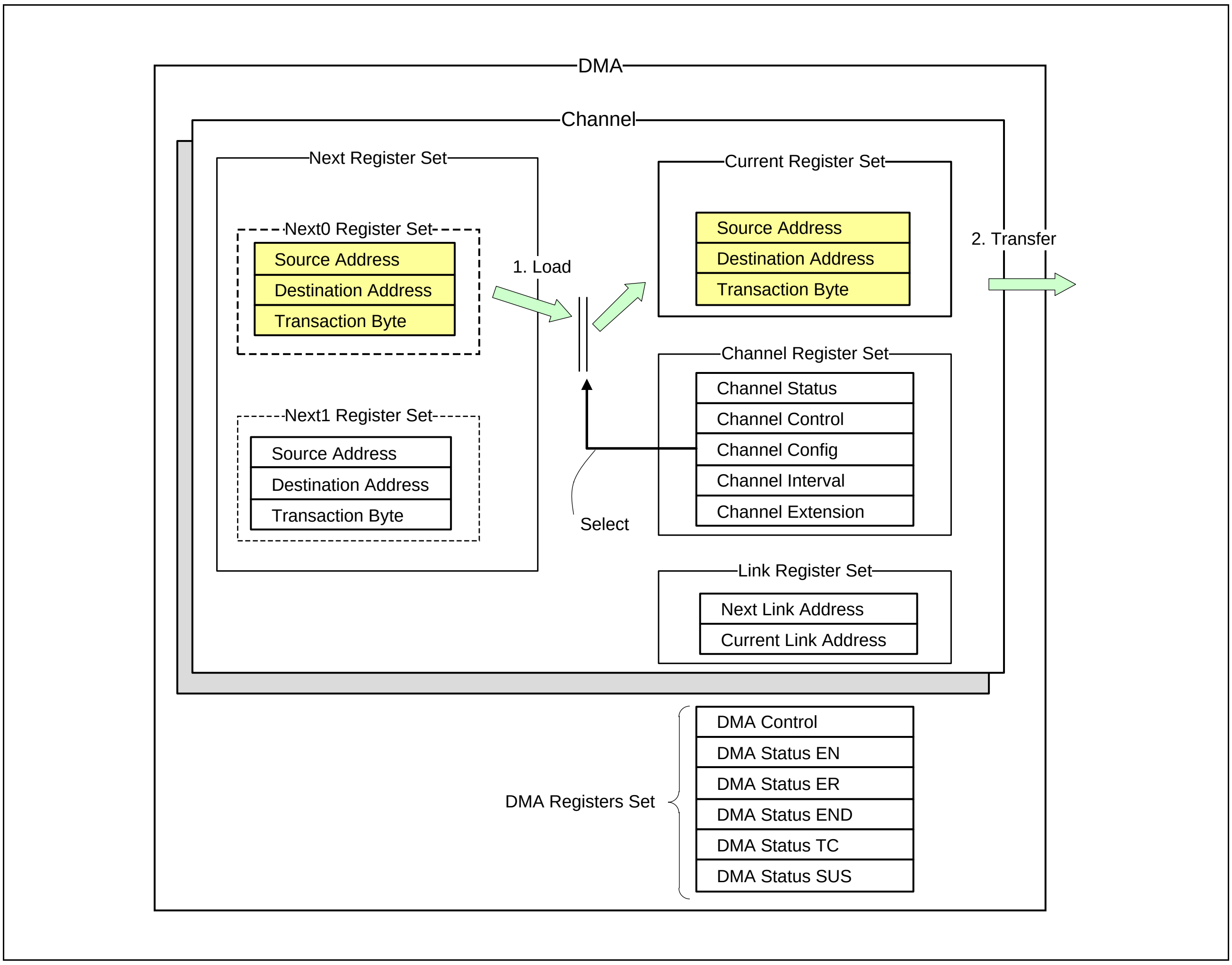


Figure 14.1 Register Configuration

(a) Next Register Set

This register set is used to set the source address, destination address, and transfer byte count of the DMA transaction to be executed next.

It consists of the Next0 register set and the Next1 register set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These registers set values are loaded to the Current Register Set and used for DMA transfer.

(b) Current Register Set

This register set indicates the source address, destination address, and transfer byte count of the currently executed DMA transaction.

The values are loaded from the Next0/1 register set (register mode) or from the descriptor read data (link mode). The user cannot write directly to this register set.

The register set is automatically updated each time a DMA transaction is executed.

(c) Channel Register Set

This register set is used to make the DMA transfer settings.
The settings to be made with this register set include channel status indication, channel control, DMA transaction setting, and DMA transaction interval.

(d) Link Register Set

This register set consists of a register that sets the address of the descriptor to be loaded next in link mode (Next Link Address Register) and a register that indicates the address of the currently executed descriptor (Current Link Address Register).

The Current Link Address Register is automatically updated when a descriptor is read. The user cannot write directly to this register set.

(e) DMA Register Set

This register set consists of a register that controls DMA as a whole and registers that indicate the status of the corresponding channels. It enables channel priority control as well as the monitoring of the channel status (EN, ER, END, TC, and SUS).

(f) Extended Resource Selector Register Set

This register set is used to select the on-chip peripheral module to perform DMA transfer request.

14.4 Register Descriptions

Table 14.1 lists the register configuration. There are eleven control registers and five status registers for each channel, and twelve common control registers are used by all channels. In addition, there is one extension resource selector per two channels.

The notation for the registers of each channel is as follows.

- Next 0 Source Address Register of channel 0 for secure access: Next 0 Source Address Register 0S (abbreviation: N0SA_0S)
- Next 0 Source Address Register of channel 0 for non-secure access: Next 0 Source Address Register 0 (abbreviation: N0SA_0)

(1) AXI I/F

Base Address Name	Base Address
<Base_S0>	H'0_1180_0000 (Cortex-A55 Address Space) H'4180_0000 (Cortex-M33 Address Space Non-Secure) H'5180_0000 (Cortex-M33 Address Space Secure)
<Base_NS0>	H'0_1182_0000 (Cortex-A55 Address Space) H'4182_0000 (Cortex-M33 Address Space Non-Secure) H'5182_0000 (Cortex-M33 Address Space Secure)

Table 14.1 Register Configuration (1/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
0	Next0 Source Address Register 0/0S	N0SA_0	N0SA_0S	RW	H'00000000	<Base_NS0>+ H'0000	<Base_S0>+ H'0000	32
	Next0 Destination Address Register 0/0S	N0DA_0	N0DA_0S	RW	H'00000000	<Base_NS0>+ H'0004	<Base_S0>+ H'0004	32
	Next0 Transaction Byte Register 0/0S	N0TB_0	N0TB_0S	RW	H'00000000	<Base_NS0>+ H'0008	<Base_S0>+ H'0008	32
	Next1 Source Address Register 0/0S	N1SA_0	N1SA_0S	RW	H'00000000	<Base_NS0>+ H'000C	<Base_S0>+ H'000C	32
	Next1 Destination Address Register 0/0S	N1DA_0	N1DA_0S	RW	H'00000000	<Base_NS0>+ H'0010	<Base_S0>+ H'0010	32
	Next1 Transaction Byte Register 0/0S	N1TB_0	N1TB_0S	RW	H'00000000	<Base_NS0>+ H'0014	<Base_S0>+ H'0014	32
	Current Source Address Register 0/0S	CRSA_0	CRSA_0S	R	H'00000000	<Base_NS0>+ H'0018	<Base_S0>+ H'0018	32
	Current Destination Address Register 0/0S	CRDA_0	CRDA_0S	R	H'00000000	<Base_NS0>+ H'001C	<Base_S0>+ H'001C	32
	Current Transaction Byte Register 0/0S	CRTB_0	CRTB_0S	R	H'00000000	<Base_NS0>+ H'0020	<Base_S0>+ H'0020	32
	Channel Status Register 0/0S	CHSTAT_0	CHSTAT_0S	R	H'00000000	<Base_NS0>+ H'0024	<Base_S0>+ H'0024	32
	Channel Control Register 0/0S	CHCTRL_0	CHCTRL_0S	RW	H'00000000	<Base_NS0>+ H'0028	<Base_S0>+ H'0028	32
	Channel Configuration Register 0/0S	CHCFG_0	CHCFG_0S	RW	H'00000000	<Base_NS0>+ H'002C	<Base_S0>+ H'002C	32
	Channel Interval Register 0/0S	CHITVL_0	CHITVL_0S	RW	H'00000000	<Base_NS0>+ H'0030	<Base_S0>+ H'0030	32
	Channel Extension Register 0/0S	CHEXT_0	CHEXT_0S	RW	H'00000000	<Base_NS0>+ H'0034	<Base_S0>+ H'0034	32

Table 14.1 Register Configuration (2/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
0	Next Link Address Register 0/0S	NXLA_0	NXLA_0S	RW	H’00000000	<Base_NS0>+H’0038	<Base_S0>+H’0038	32
	Current Link Address Register 0/0S	CRLA_0	CRLA_0S	R	H’00000000	<Base_NS0>+H’003C	<Base_S0>+H’003C	32
1	Next0 Source Address Register 1/1S	N0SA_1	N0SA_1S	RW	H’00000000	<Base_NS0>+H’0040	<Base_S0>+H’0040	32
	Next0 Destination Address Register 1/1S	N0DA_1	N0DA_1S	RW	H’00000000	<Base_NS0>+H’0044	<Base_S0>+H’0044	32
	Next0 Transaction Byte Register 1/1S	N0TB_1	N0TB_1S	RW	H’00000000	<Base_NS0>+H’0048	<Base_S0>+H’0048	32
	Next1 Source Address Register 1/1S	N1SA_1	N1SA_1S	RW	H’00000000	<Base_NS0>+H’004C	<Base_S0>+H’004C	32
	Next1 Destination Address Register 1/1S	N1DA_1	N1DA_1S	RW	H’00000000	<Base_NS0>+H’0050	<Base_S0>+H’0050	32
	Next1 Transaction Byte Register 1/1S	N1TB_1	N1TB_1S	RW	H’00000000	<Base_NS0>+H’0054	<Base_S0>+H’0054	32
	Current Source Address Register 1/1S	CRSA_1	CRSA_1S	R	H’00000000	<Base_NS0>+H’0058	<Base_S0>+H’0058	32
	Current Destination Address Register 1/1S	CRDA_1	CRDA_1S	R	H’00000000	<Base_NS0>+H’005C	<Base_S0>+H’005C	32
	Current Transaction Byte Register 1/1S	CRTB_1	CRTB_1S	R	H’00000000	<Base_NS0>+H’0060	<Base_S0>+H’0060	32
	Channel Status Register 1/1S	CHSTAT_1	CHSTAT_1S	R	H’00000000	<Base_NS0>+H’0064	<Base_S0>+H’0064	32
	Channel Control Register 1/1S	CHCTRL_1	CHCTRL_1S	RW	H’00000000	<Base_NS0>+H’0068	<Base_S0>+H’0068	32
	Channel Configuration Register 1/1S	CHCFG_1	CHCFG_1S	RW	H’00000000	<Base_NS0>+H’006C	<Base_S0>+H’006C	32
	Channel Interval Register 1/1S	CHITVL_1	CHITVL_1S	RW	H’00000000	<Base_NS0>+H’0070	<Base_S0>+H’0070	32
	Channel Extension Register 1/1S	CHEXT_1	CHEXT_1S	RW	H’00000000	<Base_NS0>+H’0074	<Base_S0>+H’0074	32
	Next Link Address Register 1/1S	NXLA_1	NXLA_1S	RW	H’00000000	<Base_NS0>+H’0078	<Base_S0>+H’0078	32
	Current Link Address Register 1/1S	CRLA_1	CRLA_1S	R	H’00000000	<Base_NS0>+H’007C	<Base_S0>+H’007C	32
2	Next0 Source Address Register 2/2S	N0SA_2	N0SA_2S	RW	H’00000000	<Base_NS0>+H’0080	<Base_S0>+H’0080	32
	Next0 Destination Address Register 2/2S	N0DA_2	N0DA_2S	RW	H’00000000	<Base_NS0>+H’0084	<Base_S0>+H’0084	32
	Next0 Transaction Byte Register 2/2S	N0TB_2	N0TB_2S	RW	H’00000000	<Base_NS0>+H’0088	<Base_S0>+H’0088	32
	Next1 Source Address Register 2/2S	N1SA_2	N1SA_2S	RW	H’00000000	<Base_NS0>+H’008C	<Base_S0>+H’008C	32
	Next1 Destination Address Register 2/2S	N1DA_2	N1DA_2S	RW	H’00000000	<Base_NS0>+H’0090	<Base_S0>+H’0090	32
	Next1 Transaction Byte Register 2/2S	N1TB_2	N1TB_2S	RW	H’00000000	<Base_NS0>+H’0094	<Base_S0>+H’0094	32
	Current Source Address Register 2/2S	CRSA_2	CRSA_2S	R	H’00000000	<Base_NS0>+H’0098	<Base_S0>+H’0098	32
	Current Destination Address Register 2/2S	CRDA_2	CRDA_2S	R	H’00000000	<Base_NS0>+H’009C	<Base_S0>+H’009C	32

Table 14.1 Register Configuration (3/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
2	Current Transaction Byte Register 2/2S	CRTB_2	CRTB_2S	R	H'00000000	<Base_NS0>+ H'00A0	<Base_S0>+ H'00A0	32
	Channel Status Register 2/2S	CHSTAT_2	CHSTAT_2S	R	H'00000000	<Base_NS0>+ H'00A4	<Base_S0>+ H'00A4	32
	Channel Control Register 2/2S	CHCTRL_2	CHCTRL_2S	RW	H'00000000	<Base_NS0>+ H'00A8	<Base_S0>+ H'00A8	32
	Channel Configuration Register 2/2S	CHCFG_2	CHCFG_2S	RW	H'00000000	<Base_NS0>+ H'00AC	<Base_S0>+ H'00AC	32
	Channel Interval Register 2/2S	CHITVL_2	CHITVL_2S	RW	H'00000000	<Base_NS0>+ H'00B0	<Base_S0>+ H'00B0	32
	Channel Extension Register 2/2S	CHEXT_2	CHEXT_2S	RW	H'00000000	<Base_NS0>+ H'00B4	<Base_S0>+ H'00B4	32
	Next Link Address Register 2/2S	NXLA_2	NXLA_2S	RW	H'00000000	<Base_NS0>+ H'00B8	<Base_S0>+ H'00B8	32
	Current Link Address Register 2/2S	CRLA_2	CRLA_2S	R	H'00000000	<Base_NS0>+ H'00BC	<Base_S0>+ H'00BC	32
3	Next0 Source Address Register 3/3S	N0SA_3	N0SA_3S	RW	H'00000000	<Base_NS0>+ H'00C0	<Base_S0>+ H'00C0	32
	Next0 Destination Address Register 3/3S	N0DA_3	N0DA_3S	RW	H'00000000	<Base_NS0>+ H'00C4	<Base_S0>+ H'00C4	32
	Next0 Transaction Byte Register 3/3S	N0TB_3	N0TB_3S	RW	H'00000000	<Base_NS0>+ H'00C8	<Base_S0>+ H'00C8	32
	Next1 Source Address Register 3/3S	N1SA_3	N1SA_3S	RW	H'00000000	<Base_NS0>+ H'00CC	<Base_S0>+ H'00CC	32
	Next1 Destination Address Register 3/3S	N1DA_3	N1DA_3S	RW	H'00000000	<Base_NS0>+ H'00D0	<Base_S0>+ H'00D0	32
	Next1 Transaction Byte Register 3/3S	N1TB_3	N1TB_3S	RW	H'00000000	<Base_NS0>+ H'00D4	<Base_S0>+ H'00D4	32
	Current Source Address Register 3/3S	CRSA_3	CRSA_3S	R	H'00000000	<Base_NS0>+ H'00D8	<Base_S0>+ H'00D8	32
	Current Destination Address Register 3/3S	CRDA_3	CRDA_3S	R	H'00000000	<Base_NS0>+ H'00DC	<Base_S0>+ H'00DC	32
	Current Transaction Byte Register 3/3S	CRTB_3	CRTB_3S	R	H'00000000	<Base_NS0>+ H'00E0	<Base_S0>+ H'00E0	32
	Channel Status Register 3/3S	CHSTAT_3	CHSTAT_3S	R	H'00000000	<Base_NS0>+ H'00E4	<Base_S0>+ H'00E4	32
	Channel Control Register 3/3S	CHCTRL_3	CHCTRL_3S	RW	H'00000000	<Base_NS0>+ H'00E8	<Base_S0>+ H'00E8	32
	Channel Configuration Register 3/3S	CHCFG_3	CHCFG_3S	RW	H'00000000	<Base_NS0>+ H'00EC	<Base_S0>+ H'00EC	32
	Channel Interval Register 3/3S	CHITVL_3	CHITVL_3S	RW	H'00000000	<Base_NS0>+ H'00F0	<Base_S0>+ H'00F0	32
	Channel Extension Register 3/3S	CHEXT_3	CHEXT_3S	RW	H'00000000	<Base_NS0>+ H'00F4	<Base_S0>+ H'00F4	32
	Next Link Address Register 3/3S	NXLA_3	NXLA_3S	RW	H'00000000	<Base_NS0>+ H'00F8	<Base_S0>+ H'00F8	32
	Current Link Address Register 3/3S	CRLA_3	CRLA_3S	R	H'00000000	<Base_NS0>+ H'00FC	<Base_S0>+ H'00FC	32

Table 14.1 Register Configuration (4/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
4	Next0 Source Address Register 4/4S	N0SA_4	N0SA_4S	RW	H'00000000	<Base_NS0>+ H'0100	<Base_S0>+ H'0100	32
	Next0 Destination Address Register 4/4S	N0DA_4	N0DA_4S	RW	H'00000000	<Base_NS0>+ H'0104	<Base_S0>+ H'0104	32
	Next0 Transaction Byte Register 4/4S	N0TB_4	N0TB_4S	RW	H'00000000	<Base_NS0>+ H'0108	<Base_S0>+ H'0108	32
	Next1 Source Address Register 4/4S	N1SA_4	N1SA_4S	RW	H'00000000	<Base_NS0>+ H'010C	<Base_S0>+ H'010C	32
	Next1 Destination Address Register 4/4S	N1DA_4	N1DA_4S	RW	H'00000000	<Base_NS0>+ H'0110	<Base_S0>+ H'0110	32
	Next1 Transaction Byte Register 4/4S	N1TB_4	N1TB_4S	RW	H'00000000	<Base_NS0>+ H'0114	<Base_S0>+ H'0114	32
	Current Source Address Register 4/4S	CRSA_4	CRSA_4S	R	H'00000000	<Base_NS0>+ H'0118	<Base_S0>+ H'0118	32
	Current Destination Address Register 4/4S	CRDA_4	CRDA_4S	R	H'00000000	<Base_NS0>+ H'011C	<Base_S0>+ H'011C	32
	Current Transaction Byte Register 4/4S	CRTB_4	CRTB_4S	R	H'00000000	<Base_NS0>+ H'0120	<Base_S0>+ H'0120	32
	Channel Status Register 4/4S	CHSTAT_4	CHSTAT_4S	R	H'00000000	<Base_NS0>+ H'0124	<Base_S0>+ H'0124	32
	Channel Control Register 4/4S	CHCTRL_4	CHCTRL_4S	RW	H'00000000	<Base_NS0>+ H'0128	<Base_S0>+ H'0128	32
	Channel Configuration Register 4/4S	CHCFG_4	CHCFG_4S	RW	H'00000000	<Base_NS0>+ H'012C	<Base_S0>+ H'012C	32
	Channel Interval Register 4/4S	CHITVL_4	CHITVL_4S	RW	H'00000000	<Base_NS0>+ H'0130	<Base_S0>+ H'0130	32
	Channel Extension Register 4/4S	CHEXT_4	CHEXT_4S	RW	H'00000000	<Base_NS0>+ H'0134	<Base_S0>+ H'0134	32
	Next Link Address Register 4/4S	NXLA_4	NXLA_4S	RW	H'00000000	<Base_NS0>+ H'0138	<Base_S0>+ H'0138	32
	Current Link Address Register 4/4S	CRLA_4	CRLA_4S	R	H'00000000	<Base_NS0>+ H'013C	<Base_S0>+ H'013C	32
5	Next0 Source Address Register 5/5S	N0SA_5	N0SA_5S	RW	H'00000000	<Base_NS0>+ H'0140	<Base_S0>+ H'0140	32
	Next0 Destination Address Register 5/5S	N0DA_5	N0DA_5S	RW	H'00000000	<Base_NS0>+ H'0144	<Base_S0>+ H'0144	32
	Next0 Transaction Byte Register 5/5S	N0TB_5	N0TB_5S	RW	H'00000000	<Base_NS0>+ H'0148	<Base_S0>+ H'0148	32
	Next1 Source Address Register 5/5S	N1SA_5	N1SA_5S	RW	H'00000000	<Base_NS0>+ H'014C	<Base_S0>+ H'014C	32
	Next1 Destination Address Register 5/5S	N1DA_5	N1DA_5S	RW	H'00000000	<Base_NS0>+ H'0150	<Base_S0>+ H'0150	32
	Next1 Transaction Byte Register 5/5S	N1TB_5	N1TB_5S	RW	H'00000000	<Base_NS0>+ H'0154	<Base_S0>+ H'0154	32
	Current Source Address Register 5/5S	CRSA_5	CRSA_5S	R	H'00000000	<Base_NS0>+ H'0158	<Base_S0>+ H'0158	32
	Current Destination Address Register 5/5S	CRDA_5	CRDA_5S	R	H'00000000	<Base_NS0>+ H'015C	<Base_S0>+ H'015C	32
	Current Transaction Byte Register 5/5S	CRTB_5	CRTB_5S	R	H'00000000	<Base_NS0>+ H'0160	<Base_S0>+ H'0160	32

Table 14.1 Register Configuration (5/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
5	Channel Status Register 5/5S	CHSTAT_5	CHSTAT_5S	R	H’00000000	<Base_NS0>+ H’0164	<Base_S0>+ H’0164	32
	Channel Control Register 5/5S	CHCTRL_5	CHCTRL_5S	RW	H’00000000	<Base_NS0>+ H’0168	<Base_S0>+ H’0168	32
	Channel Configuration Register 5/5S	CHCFG_5	CHCFG_5S	RW	H’00000000	<Base_NS0>+ H’016C	<Base_S0>+ H’016C	32
	Channel Interval Register 5/5S	CHITVL_5	CHITVL_5S	RW	H’00000000	<Base_NS0>+ H’0170	<Base_S0>+ H’0170	32
	Channel Extension Register 5/5S	CHEXT_5	CHEXT_5S	RW	H’00000000	<Base_NS0>+ H’0174	<Base_S0>+ H’0174	32
	Next Link Address Register 5/5S	NXLA_5	NXLA_5S	RW	H’00000000	<Base_NS0>+ H’0178	<Base_S0>+ H’0178	32
	Current Link Address Register 5/5S	CRLA_5	CRLA_5S	R	H’00000000	<Base_NS0>+ H’017C	<Base_S0>+ H’017C	32
6	Next0 Source Address Register 6/6S	N0SA_6	N0SA_6S	RW	H’00000000	<Base_NS0>+ H’0180	<Base_S0>+ H’0180	32
	Next0 Destination Address Register 6/6S	N0DA_6	N0DA_6S	RW	H’00000000	<Base_NS0>+ H’0184	<Base_S0>+ H’0184	32
	Next0 Transaction Byte Register 6/6S	N0TB_6	N0TB_6S	RW	H’00000000	<Base_NS0>+ H’0188	<Base_S0>+ H’0188	32
	Next1 Source Address Register 6/6S	N1SA_6	N1SA_6S	RW	H’00000000	<Base_NS0>+ H’018C	<Base_S0>+ H’018C	32
	Next1 Destination Address Register 6/6S	N1DA_6	N1DA_6S	RW	H’00000000	<Base_NS0>+ H’0190	<Base_S0>+ H’0190	32
	Next1 Transaction Byte Register 6/6S	N1TB_6	N1TB_6S	RW	H’00000000	<Base_NS0>+ H’0194	<Base_S0>+ H’0194	32
	Current Source Address Register 6/6S	CRSA_6	CRSA_6S	R	H’00000000	<Base_NS0>+ H’0198	<Base_S0>+ H’0198	32
	Current Destination Address Register 6/6S	CRDA_6	CRDA_6S	R	H’00000000	<Base_NS0>+ H’019C	<Base_S0>+ H’019C	32
	Current Transaction Byte Register 6/6S	CRTB_6	CRTB_6S	R	H’00000000	<Base_NS0>+ H’01A0	<Base_S0>+ H’01A0	32
	Channel Status Register 6/6S	CHSTAT_6	CHSTAT_6S	R	H’00000000	<Base_NS0>+ H’01A4	<Base_S0>+ H’01A4	32
	Channel Control Register 6/6S	CHCTRL_6	CHCTRL_6S	RW	H’00000000	<Base_NS0>+ H’01A8	<Base_S0>+ H’01A8	32
	Channel Configuration Register 6/6S	CHCFG_6	CHCFG_6S	RW	H’00000000	<Base_NS0>+ H’01AC	<Base_S0>+ H’01AC	32
	Channel Interval Register 6/6S	CHITVL_6	CHITVL_6S	RW	H’00000000	<Base_NS0>+ H’01B0	<Base_S0>+ H’01B0	32
	Channel Extension Register 6/6S	CHEXT_6	CHEXT_6S	RW	H’00000000	<Base_NS0>+ H’01B4	<Base_S0>+ H’01B4	32
	Next Link Address Register 6/6S	NXLA_6	NXLA_6S	RW	H’00000000	<Base_NS0>+ H’01B8	<Base_S0>+ H’01B8	32
	Current Link Address Register 6/6S	CRLA_6	CRLA_6S	R	H’00000000	<Base_NS0>+ H’01BC	<Base_S0>+ H’01BC	32

Table 14.1 Register Configuration (6/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
7	Next0 Source Address Register 7/7S	N0SA_7	N0SA_7S	RW	H'00000000	<Base_NS0>+H'01C0	<Base_S0>+H'01C0	32
	Next0 Destination Address Register 7/7S	N0DA_7	N0DA_7S	RW	H'00000000	<Base_NS0>+H'01C4	<Base_S0>+H'01C4	32
	Next0 Transaction Byte Register 7/7S	N0TB_7	N0TB_7S	RW	H'00000000	<Base_NS0>+H'01C8	<Base_S0>+H'01C8	32
	Next1 Source Address Register 7/7S	N1SA_7	N1SA_7S	RW	H'00000000	<Base_NS0>+H'01CC	<Base_S0>+H'01CC	32
	Next1 Destination Address Register 7/7S	N1DA_7	N1DA_7S	RW	H'00000000	<Base_NS0>+H'01D0	<Base_S0>+H'01D0	32
	Next1 Transaction Byte Register 7/7S	N1TB_7	N1TB_7S	RW	H'00000000	<Base_NS0>+H'01D4	<Base_S0>+H'01D4	32
	Current Source Address Register 7/7S	CRSA_7	CRSA_7S	R	H'00000000	<Base_NS0>+H'01D8	<Base_S0>+H'01D8	32
	Current Destination Address Register 7/7S	CRDA_7	CRDA_7S	R	H'00000000	<Base_NS0>+H'01DC	<Base_S0>+H'01DC	32
	Current Transaction Byte Register 7/7S	CRTB_7	CRTB_7S	R	H'00000000	<Base_NS0>+H'01E0	<Base_S0>+H'01E0	32
	Channel Status Register 7/7S	CHSTAT_7	CHSTAT_7S	R	H'00000000	<Base_NS0>+H'01E4	<Base_S0>+H'01E4	32
	Channel Control Register 7/7S	CHCTRL_7	CHCTRL_7S	RW	H'00000000	<Base_NS0>+H'01E8	<Base_S0>+H'01E8	32
	Channel Configuration Register 7/7S	CHCFG_7	CHCFG_7S	RW	H'00000000	<Base_NS0>+H'01EC	<Base_S0>+H'01EC	32
	Channel Interval Register 7/7S	CHITVL_7	CHITVL_7S	RW	H'00000000	<Base_NS0>+H'01F0	<Base_S0>+H'01F0	32
	Channel Extension Register 7/7S	CHEXT_7	CHEXT_7S	RW	H'00000000	<Base_NS0>+H'01F4	<Base_S0>+H'01F4	32
	Next Link Address Register 7/7S	NXLA_7	NXLA_7S	RW	H'00000000	<Base_NS0>+H'01F8	<Base_S0>+H'01F8	32
	Current Link Address Register 7/7S	CRLA_7	CRLA_7S	R	H'00000000	<Base_NS0>+H'01FC	<Base_S0>+H'01FC	32
Common for 0 to 7	DMA Control Registers 0-7/0-7S	DCTRL_0_7	DCTRL_0_7S	RW	H'00000000	<Base_NS0>+H'0300	<Base_S0>+H'0300	32
	DMA Status EN Registers 0-7/0-7S	DSTAT_EN_0_7	DSTAT_EN_0_7S	R	H'00000000	<Base_NS0>+H'0310	<Base_S0>+H'0310	32
	DMA Status ER Registers 0-7/0-7S	DSTAT_ER_0_7	DSTAT_ER_0_7S	R	H'00000000	<Base_NS0>+H'0314	<Base_S0>+H'0314	32
	DMA Status END Registers 0-7/0-7S	DSTAT_END_0_7	DSTAT_END_0_7S	R	H'00000000	<Base_NS0>+H'0318	<Base_S0>+H'0318	32
	DMA Status TC Registers 0-7/0-7S	DSTAT_TC_0_7	DSTAT_TC_0_7S	R	H'00000000	<Base_NS0>+H'031C	<Base_S0>+H'031C	32
	DMA Status SUS Registers 0-7/0-7S	DSTAT_SUS_0_7	DSTAT_SUS_0_7S	R	H'00000000	<Base_NS0>+H'0320	<Base_S0>+H'0320	32

Table 14.1 Register Configuration (7/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
8	Next0 Source Address Register 8/8S	N0SA_8	N0SA_8S	RW	H’00000000	<Base_NS0>+H’0400	<Base_S0>+H’0400	32
	Next0 Destination Address Register 8/8S	N0DA_8	N0DA_8S	RW	H’00000000	<Base_NS0>+H’0404	<Base_S0>+H’0404	32
	Next0 Transaction Byte Register 8/8S	N0TB_8	N0TB_8S	RW	H’00000000	<Base_NS0>+H’0408	<Base_S0>+H’0408	32
	Next1 Source Address Register 8/8S	N1SA_8	N1SA_8S	RW	H’00000000	<Base_NS0>+H’040C	<Base_S0>+H’040C	32
	Next1 Destination Address Register 8/8S	N1DA_8	N1DA_8S	RW	H’00000000	<Base_NS0>+H’0410	<Base_S0>+H’0410	32
	Next1 Transaction Byte Register 8/8S	N1TB_8	N1TB_8S	RW	H’00000000	<Base_NS0>+H’0414	<Base_S0>+H’0414	32
	Current Source Address Register 8/8S	CRSA_8	CRSA_8S	R	H’00000000	<Base_NS0>+H’0418	<Base_S0>+H’0418	32
	Current Destination Address Register 8/8S	CRDA_8	CRDA_8S	R	H’00000000	<Base_NS0>+H’041C	<Base_S0>+H’041C	32
	Current Transaction Byte Register 8/8S	CRTB_8	CRTB_8S	R	H’00000000	<Base_NS0>+H’0420	<Base_S0>+H’0420	32
	Channel Status Register 8/8S	CHSTAT_8	HSTAT_8S	R	H’00000000	<Base_NS0>+H’0424	<Base_S0>+H’0424	32
	Channel Control Register 8/8S	CHCTRL_8	CHCTRL_8S	RW	H’00000000	<Base_NS0>+H’0428	<Base_S0>+H’0428	32
	Channel Configuration Register 8/8S	CHCFG_8	CHCFG_8S	RW	H’00000000	<Base_NS0>+H’042C	<Base_S0>+H’042C	32
	Channel Interval Register 8/8S	CHITVL_8	CHITVL_8S	RW	H’00000000	<Base_NS0>+H’0430	<Base_S0>+H’0430	32
	Channel Extension Register 8/8S	CHEXT_8	CHEXT_8S	RW	H’00000000	<Base_NS0>+H’0434	<Base_S0>+H’0434	32
	Next Link Address Register 8/8S	NXLA_8	NXLA_8S	RW	H’00000000	<Base_NS0>+H’0438	<Base_S0>+H’0438	32
	Current Link Address Register 8/8S	CRLA_8	CRLA_8S	R	H’00000000	<Base_NS0>+H’043C	<Base_S0>+H’043C	32
9	Next0 Source Address Register 9/9S	N0SA_9	N0SA_9S	RW	H’00000000	<Base_NS0>+H’0440	<Base_S0>+H’0440	32
	Next0 Destination Address Register 9/9S	N0DA_9	N0DA_9S	RW	H’00000000	<Base_NS0>+H’0444	<Base_S0>+H’0444	32
	Next0 Transaction Byte Register 9/9S	N0TB_9	N0TB_9S	RW	H’00000000	<Base_NS0>+H’0448	<Base_S0>+H’0448	32
	Next1 Source Address Register 9/9S	N1SA_9	N1SA_9S	RW	H’00000000	<Base_NS0>+H’044C	<Base_S0>+H’044C	32
	Next1 Destination Address Register 9/9S	N1DA_9	N1DA_9S	RW	H’00000000	<Base_NS0>+H’0450	<Base_S0>+H’0450	32
	Next1 Transaction Byte Register 9/9S	N1TB_9	N1TB_9S	RW	H’00000000	<Base_NS0>+H’0454	<Base_S0>+H’0454	32
	Current Source Address Register 9/9S	CRSA_9	CRSA_9S	R	H’00000000	<Base_NS0>+H’0458	<Base_S0>+H’0458	32
	Current Destination Address Register 9/9S	CRDA_9	CRDA_9S	R	H’00000000	<Base_NS0>+H’045C	<Base_S0>+H’045C	32
	Current Transaction Byte Register 9/9S	CRTB_9	CRTB_9S	R	H’00000000	<Base_NS0>+H’0460	<Base_S0>+H’0460	32
	Channel Status Register 9/9S	CHSTAT_9	CHSTAT_9S	R	H’00000000	<Base_NS0>+H’0464	<Base_S0>+H’0464	32

Table 14.1 Register Configuration (8/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
9	Channel Control Register 9/9S	CHCTRL_9	CHCTRL_9S	RW	H’00000000	<Base_NS0>+H’0468	<Base_S0>+H’0468	32
	Channel Configuration Register 9/9S	CHCFG_9	CHCFG_9S	RW	H’00000000	<Base_NS0>+H’046C	<Base_S0>+H’046C	32
	Channel Interval Register 9/9S	CHITVL_9	CHITVL_9S	RW	H’00000000	<Base_NS0>+H’0470	<Base_S0>+H’0470	32
	Channel Extension Register 9/9S	CHEXT_9	CHEXT_9S	RW	H’00000000	<Base_NS0>+H’0474	<Base_S0>+H’0474	32
	Next Link Address Register 9/9S	NXLA_9	NXLA_9S	RW	H’00000000	<Base_NS0>+H’0478	<Base_S0>+H’0478	32
	Current Link Address Register 9/9S	CRLA_9	CRLA_9S	R	H’00000000	<Base_NS0>+H’047C	<Base_S0>+H’047C	32
10	Next0 Source Address Register 10/10S	N0SA_10	N0SA_10S	RW	H’00000000	<Base_NS0>+H’0480	<Base_S0>+H’0480	32
	Next0 Destination Address Register 10/10S	N0DA_10	N0DA_10S	RW	H’00000000	<Base_NS0>+H’0484	<Base_S0>+H’0484	32
	Next0 Transaction Byte Register 10/10S	N0TB_10	N0TB_10S	RW	H’00000000	<Base_NS0>+H’0488	<Base_S0>+H’0488	32
	Next1 Source Address Register 10/10S	N1SA_10	N1SA_10S	RW	H’00000000	<Base_NS0>+H’048C	<Base_S0>+H’048C	32
	Next1 Destination Address Register 10/10S	N1DA_10	N1DA_10S	RW	H’00000000	<Base_NS0>+H’0490	<Base_S0>+H’0490	32
	Next1 Transaction Byte Register 10/10S	N1TB_10	N1TB_10S	RW	H’00000000	<Base_NS0>+H’0494	<Base_S0>+H’0494	32
	Current Source Address Register 10/10S	CRSA_10	CRSA_10S	R	H’00000000	<Base_NS0>+H’0498	<Base_S0>+H’0498	32
	Current Destination Address Register 10/10S	CRDA_10	CRDA_10S	R	H’00000000	<Base_NS0>+H’049C	<Base_S0>+H’049C	32
	Current Transaction Byte Register 10/10S	CRTB_10	CRTB_10S	R	H’00000000	<Base_NS0>+H’04A0	<Base_S0>+H’04A0	32
	Channel Status Register 10/10S	CHSTAT_10	CHSTAT_10S	R	H’00000000	<Base_NS0>+H’04A4	<Base_S0>+H’04A4	32
	Channel Control Register 10/10S	CHCTRL_10	CHCTRL_10S	RW	H’00000000	<Base_NS0>+H’04A8	<Base_S0>+H’04A8	32
	Channel Configuration Register 10/10S	CHCFG_10	CHCFG_10S	RW	H’00000000	<Base_NS0>+H’04AC	<Base_S0>+H’04AC	32
	Channel Interval Register 10/10S	CHITVL_10	CHITVL_10S	RW	H’00000000	<Base_NS0>+H’04B0	<Base_S0>+H’04B0	32
	Channel Extension Register 10/10S	CHEXT_10	CHEXT_10S	RW	H’00000000	<Base_NS0>+H’04B4	<Base_S0>+H’04B4	32
	Next Link Address Register 10/10S	NXLA_10	NXLA_10S	RW	H’00000000	<Base_NS0>+H’04B8	<Base_S0>+H’04B8	32
	Current Link Address Register 10/10S	CRLA_10	CRLA_10S	R	H’00000000	<Base_NS0>+H’04BC	<Base_S0>+H’04BC	32

Table 14.1 Register Configuration (9/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
11	Next0 Source Address Register 11/11S	N0SA_11	N0SA_11S	RW	H'00000000	<Base_NS0> +H'04C0	<Base_S0>+ H'04C0	32
	Next0 Destination Address Register 11/11S	N0DA_11	N0DA_11S	RW	H'00000000	<Base_NS0> +H'04C4	<Base_S0>+ H'04C4	32
	Next0 Transaction Byte Register 11/11S	N0TB_11	N0TB_11S	RW	H'00000000	<Base_NS0> +H'04C8	<Base_S0>+ H'04C8	32
	Next1 Source Address Register 11/11S	N1SA_11	N1SA_11S	RW	H'00000000	<Base_NS0> +H'04CC	<Base_S0>+ H'04CC	32
	Next1 Destination Address Register 11/11S	N1DA_11	N1DA_11S	RW	H'00000000	<Base_NS0> +H'04D0	<Base_S0>+ H'04D0	32
	Next1 Transaction Byte Register 11/11S	N1TB_11	N1TB_11S	RW	H'00000000	<Base_NS0> +H'04D4	<Base_S0>+ H'04D4	32
	Current Source Address Register 11/11S	CRSA_11	CRSA_11S	R	H'00000000	<Base_NS0> +H'04D8	<Base_S0>+ H'04D8	32
	Current Destination Address Register 11/11S	CRDA_11	CRDA_11S	R	H'00000000	<Base_NS0> +H'04DC	<Base_S0>+ H'04DC	32
	Current Transaction Byte Register 11/11S	CRTB_11	CRTB_11S	R	H'00000000	<Base_NS0> +H'04E0	<Base_S0>+ H'04E0	32
	Channel Status Register 11/11S	CHSTAT_11	CHSTAT_11S	R	H'00000000	<Base_NS0> +H'04E4	<Base_S0>+ H'04E4	32
	Channel Control Register 11/11S	CHCTRL_11	CHCTRL_11S	RW	H'00000000	<Base_NS0> +H'04E8	<Base_S0>+ H'04E8	32
	Channel Configuration Register 11/11S	CHCFG_11	CHCFG_11S	RW	H'00000000	<Base_NS0> +H'04EC	<Base_S0>+ H'04EC	32
	Channel Interval Register 11/11S	CHITVL_11	CHITVL_11S	RW	H'00000000	<Base_NS0> +H'04F0	<Base_S0>+ H'04F0	32
	Channel Extension Register 11/11S	CHEXT_11	CHEXT_11S	RW	H'00000000	<Base_NS0> +H'04F4	<Base_S0>+ H'04F4	32
	Next Link Address Register 11/11S	NXLA_11	NXLA_11S	RW	H'00000000	<Base_NS0> +H'04F8	<Base_S0>+ H'04F8	32
	Current Link Address Register 11/11S	CRLA_11	CRLA_11S	R	H'00000000	<Base_NS0> +H'04FC	<Base_S0>+ H'04FC	32
12	Next0 Source Address Register 12/12S	N0SA_12	N0SA_12S	RW	H'00000000	<Base_NS0> +H'0500	<Base_S0>+ H'0500	32
	Next0 Destination Address Register 12/12S	N0DA_12	N0DA_12S	RW	H'00000000	<Base_NS0> +H'0504	<Base_S0>+ H'0504	32
	Next0 Transaction Byte Register 12/12S	N0TB_12	N0TB_12S	RW	H'00000000	<Base_NS0> +H'0508	<Base_S0>+ H'0508	32
	Next1 Source Address Register 12/12S	N1SA_12	N1SA_12S	RW	H'00000000	<Base_NS0> +H'050C	<Base_S0>+ H'050C	32
	Next1 Destination Address Register 12/12S	N1DA_12	N1DA_12S	RW	H'00000000	<Base_NS0> +H'0510	<Base_S0>+ H'0510	32
	Next1 Transaction Byte Register 12/12S	N1TB_12	N1TB_12S	RW	H'00000000	<Base_NS0> +H'0514	<Base_S0>+ H'0514	32
	Current Source Address Register 12/12S	CRSA_12	CRSA_12S	R	H'00000000	<Base_NS0> +H'0518	<Base_S0>+ H'0518	32
	Current Destination Address Register 12/12S	CRDA_12	CRDA_12S	R	H'00000000	<Base_NS0> +H'051C	<Base_S0>+ H'051C	32
	Current Transaction Byte Register 12/12S	CRTB_12	CRTB_12S	R	H'00000000	<Base_NS0> +H'0520	<Base_S0>+ H'0520	32
	Channel Status Register 12/12S	CHSTAT_12	CHSTAT_12S	R	H'00000000	<Base_NS0> +H'0524	<Base_S0>+ H'0524	32

Table 14.1 Register Configuration (10/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
12	Channel Control Register 12/12S	CHCTRL_12	CHCTRL_12S	RW	H’00000000	<Base_NS0>+H’0528	<Base_S0>+H’0528	32
	Channel Configuration Register 12/12S	CHCFG_12	CHCFG_12S	RW	H’00000000	<Base_NS0>+H’052C	<Base_S0>+H’052C	32
	Channel Interval Register 12/12S	CHITVL_12	CHITVL_12S	RW	H’00000000	<Base_NS0>+H’0530	<Base_S0>+H’0530	32
	Channel Extension Register 12/12S	CHEXT_12	CHEXT_12S	RW	H’00000000	<Base_NS0>+H’0534	<Base_S0>+H’0534	32
	Next Link Address Register 12/12S	NXLA_12	NXLA_12S	RW	H’00000000	<Base_NS0>+H’0538	<Base_S0>+H’0538	32
	Current Link Address Register 12/12S	CRLA_12	CRLA_12S	R	H’00000000	<Base_NS0>+H’053C	<Base_S0>+H’053C	32
13	Next0 Source Address Register 13/13S	N0SA_13	N0SA_13S	RW	H’00000000	<Base_NS0>+H’0540	<Base_S0>+H’0540	32
	Next0 Destination Address Register 13/13S	N0DA_13	N0DA_13S	RW	H’00000000	<Base_NS0>+H’0544	<Base_S0>+H’0544	32
	Next0 Transaction Byte Register 13/13S	N0TB_13	N0TB_13S	RW	H’00000000	<Base_NS0>+H’0548	<Base_S0>+H’0548	32
	Next1 Source Address Register 13/13S	N1SA_13	N1SA_13S	RW	H’00000000	<Base_NS0>+H’054C	<Base_S0>+H’054C	32
	Next1 Destination Address Register 13/13S	N1DA_13	N1DA_13S	RW	H’00000000	<Base_NS0>+H’0550	<Base_S0>+H’0550	32
	Next1 Transaction Byte Register 13/13S	N1TB_13	N1TB_13S	RW	H’00000000	<Base_NS0>+H’0554	<Base_S0>+H’0554	32
	Current Source Address Register 13/13S	CRSA_13	CRSA_13S	R	H’00000000	<Base_NS0>+H’0558	<Base_S0>+H’0558	32
	Current Destination Address Register 13/13S	CRDA_13	CRDA_13S	R	H’00000000	<Base_NS0>+H’055C	<Base_S0>+H’055C	32
	Current Transaction Byte Register 13/13S	CRTB_13	CRTB_13S	R	H’00000000	<Base_NS0>+H’0560	<Base_S0>+H’0560	32
	Channel Status Register 13/13S	CHSTAT_13	CHSTAT_13S	R	H’00000000	<Base_NS0>+H’0564	<Base_S0>+H’0564	32
	Channel Control Register 13/13S	CHCTRL_13	CHCTRL_13S	RW	H’00000000	<Base_NS0>+H’0568	<Base_S0>+H’0568	32
	Channel Configuration Register 13/13S	CHCFG_13	CHCFG_13S	RW	H’00000000	<Base_NS0>+H’056C	<Base_S0>+H’056C	32
	Channel Interval Register 13/13S	CHITVL_13	CHITVL_13S	RW	H’00000000	<Base_NS0>+H’0570	<Base_S0>+H’0570	32
	Channel Extension Register 13/13S	CHEXT_13	CHEXT_13S	RW	H’00000000	<Base_NS0>+H’0574	<Base_S0>+H’0574	32
	Next Link Address Register 13/13S	NXLA_13	NXLA_13S	RW	H’00000000	<Base_NS0>+H’0578	<Base_S0>+H’0578	32
	Current Link Address Register 13/13S	CRLA_13	CRLA_13S	R	H’00000000	<Base_NS0>+H’057C	<Base_S0>+H’057C	32

Table 14.1 Register Configuration (11/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
14	Next0 Source Address Register 14/14S	N0SA_14	N0SA_14S	RW	H'00000000	<Base_NS0> +H'0580	<Base_S0>+ H'0580	32
	Next0 Destination Address Register 14/14S	N0DA_14	N0DA_14S	RW	H'00000000	<Base_NS0> +H'0584	<Base_S0>+ H'0584	32
	Next0 Transaction Byte Register 14/14S	N0TB_14	N0TB_14S	RW	H'00000000	<Base_NS0> +H'0588	<Base_S0>+ H'0588	32
	Next1 Source Address Register 14/14S	N1SA_14	N1SA_14S	RW	H'00000000	<Base_NS0> +H'058C	<Base_S0>+ H'058C	32
	Next1 Destination Address Register 14/14S	N1DA_14	N1DA_14S	RW	H'00000000	<Base_NS0> +H'0590	<Base_S0>+ H'0590	32
	Next1 Transaction Byte Register 14/14S	N1TB_14	N1TB_14S	RW	H'00000000	<Base_NS0> +H'0594	<Base_S0>+ H'0594	32
	Current Source Address Register 14/14S	CRSA_14	CRSA_14S	R	H'00000000	<Base_NS0> +H'0598	<Base_S0>+ H'0598	32
	Current Destination Address Register 14/14S	CRDA_14	CRDA_14S	R	H'00000000	<Base_NS0> +H'059C	<Base_S0>+ H'059C	32
	Current Transaction Byte Register 14/14S	CRTB_14	CRTB_14S	R	H'00000000	<Base_NS0> +H'05A0	<Base_S0>+ H'05A0	32
	Channel Status Register 14/14S	CHSTAT_14	CHSTAT_14S	R	H'00000000	<Base_NS0> +H'05A4	<Base_S0>+ H'05A4	32
	Channel Control Register 14/14S	CHCTRL_14	CHCTRL_14S	RW	H'00000000	<Base_NS0> +H'05A8	<Base_S0>+ H'05A8	32
	Channel Configuration Register 14/14S	CHCFG_14	CHCFG_14S	RW	H'00000000	<Base_NS0> +H'05AC	<Base_S0>+ H'05AC	32
	Channel Interval Register 14/14S	CHITVL_14	CHITVL_14S	RW	H'00000000	<Base_NS0> +H'05B0	<Base_S0>+ H'05B0	32
	Channel Extension Register 14/14S	CHEXT_14	CHEXT_14S	RW	H'00000000	<Base_NS0> +H'05B4	<Base_S0>+ H'05B4	32
	Next Link Address Register 14/14S	NXLA_14	NXLA_14S	RW	H'00000000	<Base_NS0> +H'05B8	<Base_S0>+ H'05B8	32
	Current Link Address Register 14/14S	CRLA_14	CRLA_14S	R	H'00000000	<Base_NS0> +H'05BC	<Base_S0>+ H'05BC	32

Table 14.1 Register Configuration (12/12)

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
15	Next0 Source Address Register 15/15S	N0SA_15	N0SA_15S	RW	H’00000000	<Base_NS0>+H’05C0	<Base_S0>+H’05C0	32
	Next0 Destination Address Register 15/15S	N0DA_15	N0DA_15S	RW	H’00000000	<Base_NS0>+H’05C4	<Base_S0>+H’05C4	32
	Next0 Transaction Byte Register 15/15S	N0TB_15	N0TB_15S	RW	H’00000000	<Base_NS0>+H’05C8	<Base_S0>+H’05C8	32
	Next1 Source Address Register 15/15S	N1SA_15	N1SA_15S	RW	H’00000000	<Base_NS0>+H’05CC	<Base_S0>+H’05CC	32
	Next1 Destination Address Register 15/15S	N1DA_15	N1DA_15S	RW	H’00000000	<Base_NS0>+H’05D0	<Base_S0>+H’05D0	32
	Next1 Transaction Byte Register 15/15S	N1TB_15	N1TB_15S	RW	H’00000000	<Base_NS0>+H’05D4	<Base_S0>+H’05D4	32
	Current Source Address Register 15/15S	CRSA_15	CRSA_15S	R	H’00000000	<Base_NS0>+H’05D8	<Base_S0>+H’05D8	32
	Current Destination Address Register 15/15S	CRDA_15	CRDA_15S	R	H’00000000	<Base_NS0>+H’05DC	<Base_S0>+H’05DC	32
	Current Transaction Byte Register 15/15S	CRTB_15	CRTB_15S	R	H’00000000	<Base_NS0>+H’05E0	<Base_S0>+H’05E0	32
	Channel Status Register 15/15S	CHSTAT_15	CHSTAT_15S	R	H’00000000	<Base_NS0>+H’05E4	<Base_S0>+H’05E4	32
	Channel Control Register 15/15S	CHCTRL_15	CHCTRL_15S	RW	H’00000000	<Base_NS0>+H’05E8	<Base_S0>+H’05E8	32
	Channel Configuration Register 15/15S	CHCFG_15	CHCFG_15S	RW	H’00000000	<Base_NS0>+H’05EC	<Base_S0>+H’05EC	32
	Channel Interval Register 15/15S	CHITVL_15	CHITVL_15S	RW	H’00000000	<Base_NS0>+H’05F0	<Base_S0>+H’05F0	32
	Channel Extension Register 15/15S	CHEXT_15	CHEXT_15S	RW	H’00000000	<Base_NS0>+H’05F4	<Base_S0>+H’05F4	32
	Next Link Address Register 15/15S	NXLA_15	NXLA_15S	RW	H’00000000	<Base_NS0>+H’05F8	<Base_S0>+H’05F8	32
	Current Link Address Register 15/15S	CRLA_15	CRLA_15S	R	H’00000000	<Base_NS0>+H’05FC	<Base_S0>+H’05FC	32
Common for 8 to 15	DMA Control Registers 8-15/8-15S	DCTRL_8_15	DCTRL_8_15S	RW	H’00000000	<Base_NS0>+H’0700	<Base_S0>+H’0700	32
	DMA Status EN Registers 8-15/8-15S	DSTAT_EN_8_15	DSTAT_EN_8_15S	R	H’00000000	<Base_NS0>+H’0710	<Base_S0>+H’0710	32
	DMA Status ER Registers 8-15/8-15S	DSTAT_ER_8_15	DSTAT_ER_8_15S	R	H’00000000	<Base_NS0>+H’0714	<Base_S0>+H’0714	32
	DMA Status END Registers 8-15/8-15S	DSTAT_END_8_15	DSTAT_END_8_15S	R	H’00000000	<Base_NS0>+H’0718	<Base_S0>+H’0718	32
	DMA Status TC Registers 8-15/8-15S	DSTAT_TC_8_15	DSTAT_TC_8_15S	R	H’00000000	<Base_NS0>+H’071C	<Base_S0>+H’071C	32
	DMA Status SUS Registers 8-15/8-15S	DSTAT_SUS_8_15	DSTAT_SUS_8_15S	R	H’00000000	<Base_NS0>+H’0720	<Base_S0>+H’0720	32

(2) APB I/F

Base Address Name	Base Address
<Base_S1>	H’0_1181_0000 (Cortex-A55 Address Space) H’4181_0000 (Cortex-M33 Address Space Non-Secure) H’5181_0000 (Cortex-M33 Address Space Secure)
<Base_NS1>	H’0_1183_0000 (Cortex-A55 Address Space) H’4183_0000 (Cortex-M33 Address Space Non-Secure) H’5183_0000 (Cortex-M33 Address Space Secure)

Table 14.2 Register Configuration

Channel	Register Name	Abbreviation		R/W	Initial Value	Address		Access Size
		Non-secure	Secure			Non-secure	Secure	
0/1	DMA Extension Resource Selectors 0/0S	DMARS0	DMARS0S	RW	H’00000000	<Base_NS1>+ H’0000	<Base_S1>+ H’0000	32
2/3	DMA Extension Resource Selectors 1/1S	DMARS1	DMARS1S	RW	H’00000000	<Base_NS1>+ H’0004	<Base_S1>+ H’0004	32
4/5	DMA Extension Resource Selectors 2/2S	DMARS 2	DMARS 2S	RW	H’00000000	<Base_NS1>+ H’0008	<Base_S1>+ H’0008	32
6/7	DMA Extension Resource Selectors 3/3S	DMARS 3	DMARS 3S	RW	H’00000000	<Base_NS1>+ H’000C	<Base_S1>+ H’000C	32
8/9	DMA Extension Resource Selectors 4/4S	DMARS 4	DMARS 4S	RW	H’00000000	<Base_NS1>+ H’0010	<Base_S1>+ H’0010	32
10/11	DMA Extension Resource Selectors 5/5S	DMARS 5	DMARS 5S	RW	H’00000000	<Base_NS1>+ H’0014	<Base_S1>+ H’0014	32
12/13	DMA Extension Resource Selectors 6/6S	DMARS 6	DMARS 6S	RW	H’00000000	<Base_NS1>+ H’0018	<Base_S1>+ H’0018	32
14/15	DMA Extension Resource Selectors 7/7S	DMARS 7	DMARS 7S	RW	H’00000000	<Base_NS1>+ H’001C	<Base_S1>+ H’001C	32

14.4.1 Next Source Address Register n/nS (N0SA_n/nS, N1SA_n/nS)

This register sets the DMA transfer source address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next. N0SA_n/nS is for the Next0 Register Set, and N1SA_n/nS is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA	All 0	R/W	Source Address Sets the start address of the DMA transfer source.

14.4.2 Next Destination Address Register n/nS (N0DA_n/nS, N1DA_n/nS)

This register sets the DMA transfer destination address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next.

N0DA_n/nS is for the Next0 Register Set, and N1DA_n/nS is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA	All 0	R/W	Destination Address Sets the start address of the DMA transfer destination.

14.4.3 Next Transaction Byte Register n/nS (N0TB_n/nS, N1TB_n/nS)

This register sets the total transfer byte count (DMA transaction) of DMA channel (n = 0 to 15) which is to be executed next.

N0TB_n/nS is for the Next0 Register Set, and N1TB_n/nS is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB	All 0	R/W	Transaction Byte Sets the total transfer byte count. *Note*: Do not start a DMA transaction with 0 set in this register.

14.4.4 Current Source Address Register n/nS (CRSA_n/nS)

This register indicates the DMA transfer source address of DMA channel n (n = 0 to 15).

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA	All 0	R	Current Source Address Register Indicates the read address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in SAD of the CHCFG_n/nS register.) The value increments when a read transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

14.4.5 Current Destination Address Register n/nS (CRDA_n/nS)

This register indicates the DMA transfer destination address of DMA channel n (n = 0 to 15).

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA	All 0	R	Current Destination Address Register Indicates the write address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in DAD of the CHCFG_n/nS register.) The value increments when a write transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

14.4.6 Current Transaction Byte Register n/nS (CRTB_n/nS)

This register indicates the total transfer byte count of DMA channel n (n = 0 to 15). The value of this register becomes 0 when the transaction ends.

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB	All 0	R	Current Transaction Byte Register Indicates the remaining transfer byte count of the currently executed DMA transaction. The value automatically decrements during the DMA transaction. The value decrements when a write transfer is completed. Read this register after DMA stops (0 is set in EN of the CHSTAT_n/nS register). (Any value obtained during the DMA operation should be handled as a reference value.)

14.4.7 Channel Status Register n/nS (CHSTAT_n/nS)

This register indicates the status of DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTMSK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
16	INTMSK	0	R	Indicates the temporary mask status of the DMA transfer end interrupt. 1: Masked temporarily 0: Unmasked temporarily Set condition(s): ● When SETINTMSK (CHCTRL_n/nS) is set to 1 Reset condition(s): ● When CLRINTMSK (CHCTRL_n/nS) is set to 1 ● When SWRST (CHCTRL_n/nS) is set to 1
15 to 12	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
11	MODE	0	R	DMA Mode Indicates the DMA mode. It corresponds to the value set in the DMS bit of the CHCFG_n/nS register. 0: Register mode 1: Link mode
10	DER	0	R	Descriptor Error Indicates whether the link valid value of the read descriptor is invalid (LV = 0) (this is not dependent on the DIM level of the descriptor). If a descriptor error has occurred, the transfer is stopped but no DMA error interrupt occurs. 0: Descriptor Error not detected 1: Descriptor Error detected Set condition(s): ● When the LV value loaded with the descriptor in link mode is 0 Reset condition(s): ● When SWRST (CHCTRL_n/nS) is set to 1
9	DW	0	R	Descriptor WriteBack Indicates the descriptor writeback status. The bit maintains 1 if a bus error is received during descriptor writeback. 0: Operation other than writeback is being performed for the header in link mode. 1: (ER = 0) Writeback is being performed for the header in link mode. (ER = 1) A bus error occurs during writeback for the header in link mode. Set condition(s): ● When header writeback in link mode starts Reset condition(s): ● When header writeback in link mode ends with an OK response ● When SWRST (CHCTRL_n/nS) is set to 1

Bit	Bit Name	Initial Value	R/W	Description
8	DL	0	R	Descriptor Load Indicates whether the descriptor is being loaded. The bit maintains 1 if a bus error is received during descriptor load. 0: Operation other than descriptor load 1: (ER = 0) Descriptor load is in progress in link mode. (ER = 1) A bus error occurs during descriptor load in link mode. Set condition(s): ● When descriptor load in link mode starts Reset condition(s): ● When descriptor load in link mode ends with an OK response ● When SWRST (CHCTRL_n/nS) is set to 1
7	SR	0	R	Selected Register Set Indicates the register set currently selected in register mode. 0: Next0 Register Set 1: Next1 Register Set Set condition(s): ● When RSEL (CHCFG_n/nS) is set to 1 Reset condition(s): ● When RSEL (CHCFG_n/nS) is set to 0
6	TC	0	R	Terminal Count Indicates whether the DMA transaction is completed. 0: DMA transfer not completed 1: DMA transfer completed Set condition(s): ● When data equivalent to the total transfer byte count set in the CRTB register has been transferred in register mode ● When data equivalent to the total transfer byte count set in the CRTB register has been transferred in link mode, with 1 set in WBD of the descriptor header ● When descriptor writeback is completed in link mode, with 0 set in WBD of the descriptor header Clear condition(s): ● When the CLRTC (CHCTRL_n/nS) bit is set to 1 ● When the SWRST (CHCTRL_n/nS) bit is set to 1
5	END	0	R	DMAEND Interrupted Indicates whether the DMA transaction is completed and whether the DMA transfer end interrupt has occurred. 0: DMA transfer not completed 1: DMA transfer completed Set condition(s): ● When one of the set conditions for the TC bit is met and 0 is set in DEM of the CHCFG_n/nS register ● When the descriptor is read in link mode and both LV of the header and DIM are set to 0 Clear condition(s): ● When CLREND (CHCTRL_n/nS) is set to 1 ● When SWRST (CHCTRL_n/nS) is set to 1

Bit	Bit Name	Initial Value	R/W	Description
4	ER	0	R	Error bit Indicates that a DMA error interrupt has occurred because an error response has been received from the transfer source or destination and a bus error has occurred during the DMA transfer. 0: No bus error has occurred 1: A DMA error interrupt has occurred due to a bus error Set condition(s): ● When a bus error has occurred during a bus cycle Clear condition(s): ● When SWRST (CHCTRL_n/nS) is set to 1
3	SUS	0	R	Suspend Indicates whether the channel is suspended. 0: Channel_n not suspended 1: Channel_n suspended Set condition(s): ● When SETSUS (CHCTRL_n/nS) is set to 1 during a DMA transfer on Channel_n, creating a SUSPEND status internally Clear condition(s): ● When CLRSUS (CHCTRL_n/nS) is set to 1 ● When CLREN (CHCTRL_n/nS) is set to 1
2	TACT	0	R	Transaction Active Indicates whether the DMAC is active. This bit is intended to check that the channel is completely inactive. 0: DMA on Channel_n inactive 1: DMA on Channel_n active Set condition(s): ● When a DMA transaction starts on Channel_n Clear condition(s): ● When a DMA transaction is completed
1	RQST	0	R	Request Indicates whether a transfer request is being received. 0: DMA transfer request not being received 1: DMA transfer request being received Set condition(s): ● When the STG bit (CHCTRL_n/nS) is set to 1 (auto request) ● When a transfer request is received from the DMA request source set in the CHCFG_n/nS register Clear condition(s): ● When SWRST (CHCTRL_n/nS) is set to 1 ● When CLRRQ (CHCTRL_n/nS) is set to 1 ● When a transfer is executed on the side specified by REQD (CHCFG_n/nS) in single transfer mode (TM = 0). ● When all DMA transactions are completed in register mode (the transaction ends with REN set to 0) ● When the DMA transfer of the last descriptor (LE = 1) is completed in link mode ● When descriptor read stops (LV = 0) in link mode ● When a bus error is received due to an error response

Bit	Bit Name	Initial Value	R/W	Description
0	EN	0	R	Enable Indicates whether the operation of DMA channel n is enabled or disabled. 0: Operation disabled 1: Operation enabled Set condition(s): ● When SETEN (CHCTRL_n/nS) is set to 1 Clear condition(s): ● When SWRST (CHCTRL_n/nS) is set to 1 ● When CLREN (CHCTRL_n/nS) is set to 1 ● When a bus error is received due to an error response during the transfer ● When all DMA transactions are completed in register mode (the transaction ends with REN set to 0) ● When the DMA transfer of the last descriptor (LE = 1) is completed in link mode (writeback when WBD is set to 0) ● When descriptor read stops (LV = 0) in link mode

CAUTION

If the ER bit is set to 1 for any transfer, the whole transfer should be handled as invalid.

To suspend a DMA transaction, mask or clear the transfer request or clear the Enable bit (for the procedure, see **Section 14.7.8(2), Transfer Stop**).

If a transfer request from an on-chip peripheral module input is made concurrently with an auto request (by setting 1 in the STG bit) for the same one channel, the trigger source that takes effect cannot be identified. Make sure that only one of these transfer requests is used in the system.

When transfer is requested by an auto request, wait for the last requested DMA transfer to complete (use the Current Register or other data to check the status) before setting the STG bit for the next transfer request.

14.4.8 Channel Control Register n/nS (CHCTRL_n/nS)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLRINT MSK	SETINT MSK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLRSU S	SETSU S	—	CLRTC	CLREN D	CLRRQ	SWRST	STG	CLREN	SETEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
17	CLRINTMSK	0	R/W	When this bit is set to 1, the mask of the DMA transfer end interrupt is cleared. Also, the INTMSK bit of the CHSTAT_n/nS register is set to 0. If the mask is cleared when 1 is set in both LVINT of the DCTRL register and END of the CHSTAT_n/nS register, the DMA transfer end interrupt becomes active. (It does not become active when 0 is set in LVINT.) A read operation results in 0 being read. 1: Clears the mask set by SETINTMSK. 0: Does not affect the operation.
16	SETINTMSK	0	R/W	When this bit is set to 1, the DMA transfer end interrupt is temporarily masked. Also, the INTMSK bit of the CHSTAT_n/nS register is set to 1. A read operation results in 0 being read. 1: Masks the DMA transfer end interrupt. 0: Does not affect the operation.
15 to 10	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
9	CLRSUS	0	R/W	Clear Suspend Clears the suspend status. Setting this bit to 1 when 1 is set in SUS of the CHSTAT_n/nS register can clear the suspend status. An attempt to read this bit results in 0 being read. 1: Clears the suspend status of the current DMA transfer. 0: Does not affect the operation.
8	SETSUS	0	R/W	Set Suspend Suspends the current DMA transfer. Setting this bit to 1 when 1 is set in EN of the CHSTAT_n/nS register can suspend the current DMA transfer. An attempt to read this bit results in 0 being read. 1: Suspends the current DMA transfer. 0: Does not affect the operation.
7	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
6	CLRTC	0	R/W	Clear TC bit Setting this bit to 1 can clear the TC bit of the CHSTAT_n/nS register. An attempt to read this bit results in 0 being read. 1: Clears the TC bit. 0: Does not affect the operation.

Bit	Bit Name	Initial Value	R/W	Description
5	CLREND	0	R/W	Clear End bit Setting this bit to 1 can clear the END bit of the CHSTAT_n/nS register. Also, the DMA transfer end interrupt is cleared. An attempt to read this bit results in 0 being read. 1: Clears the END bit. 0: Does not affect the operation.
4	CLRRQ	0	R/W	Clear Request bit Setting this bit to 1 can clear the RQST bit of the CHSTAT_n/nS register. An attempt to read this bit results in 0 being read. 1: Clears the RQST bit. 0: Does not affect the operation.
3	SWRST	0	R/W	Software Reset Setting this bit to 1 can clear the channel status register (CHSTAT_n/nS). When setting this bit to 1, make sure that both the EN bit and TACT bit are set to 0. An attempt to read this bit results in 0 being read. 1: Resets the channel status register. 0: Does not affect the operation.
2	STG	0	R/W	Software Trigger Setting this bit to 1 sets an auto request. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence. An attempt to read this bit results in 0 being read. 1: Sets a transfer request triggered by an auto request (sets 1 in the RQST bit). 0: Does not affect the operation.
1	CLREN	0	R/W	Clear Enable Setting this bit to 1 can clear the EN bit (for details, see **Section 14.7.8(2), Transfer Stop**). An attempt to read this bit results in 0 being read. 1: Stops the DMA transfer (clears the EN bit). 0: Does not affect the operation.
0	SETEN	0	R/W	Set Enable Enables a DMA transfer on DMA channel n. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence and the transfer does not start. An attempt to read this bit results in 0 being read. 1: Enables a DMA transfer (sets 1 in the EN bit). 0: Does not affect the operation.

14.4.9 Channel Configuration Register n/nS (CHCFG_n/nS)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	—	—	DEM	—	TM	DAD	SAD	DDS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS[3:0]				—	AM[2:0]			—	LVL	HIEN	LOEN	REQD	SEL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0	R/W	DMA Mode Select Sets the DMA mode. 0: Register mode (initial value) 1: Link mode
30	REN	0	R/W	Register Set Enable After a DMA transaction is completed, DMA transfers are continued using the Next register set selected by RSEL. This bit is valid only in register mode. 0: Does not continue DMA transfers. 1: Continues DMA transfers. Set condition(s): ● When 1 is written to this bit Clear condition(s): ● When 0 is written to this bit ● When a DMA transaction is completed, with REN set to 1
29	RSW	0	R/W	Register Select Switch Inverts RSEL automatically after a DMA transaction is completed. This bit is valid only in register mode. 0: Does not invert RSEL automatically after a DMA transaction (initial value). 1: Inverts RSEL automatically after a DMA transaction.
28	RSEL	0	R/W	Register Set Select Selects the Next register set to be executed next. This bit is valid only in register mode. When RSW is set to 1, this bit is inverted automatically when a DMA transaction is completed. 0: Executes the Next0 Register Set (initial value). 1: Executes the Next1 Register Set. Transition condition(s): ● When a DMA transaction is completed, with RSW set to 1
27	SBE	0	R/W	Sweep Buffer Enable Selects whether to sweep (write) the data already read into the buffer and stop the DMA transfer if the Enable bit is cleared to 0 during a DMA transaction. The sweep mode is available only when REQD is set to 0. 0: Stops the DMA transfer without sweeping the buffer (initial value). 1: Stops the DMA transfer after sweeping the buffer.
26, 25	—	0	R	Reserved. Set 0. A read operation results in 0 being read.

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>24</td><td>DEM</td><td>0</td><td>R/W</td><td>DMA Transfer End Interrupt Mask
Masks the DMA transfer end interrupt for register mode transfer.
If 1 is set in this bit when a DMA transfer end interrupt is output, the DMA transfer end interrupt signal is not asserted. In this case, DEM is cleared to 0 automatically.
0: Does not mask the DMA transfer end interrupt (initial value).
1: Masks the DMA transfer end interrupt.
Clear condition(s):
● When a DMA transaction is completed with DEM set to 1</td></tr>
<tr><td>23</td><td>—</td><td>0</td><td>R</td><td>Reserved. Set 0. A read operation results in 0 being read.</td></tr>
<tr><td>22</td><td>TM</td><td>0</td><td>R/W</td><td>Transfer Mode
Sets the DMA transfer mode.
0: Single transfer mode (initial value)
1: Block transfer mode</td></tr>
<tr><td>21</td><td>DAD</td><td>0</td><td>R/W</td><td>Sets the destination address counting direction of DMA channel n.
0: Increment (initial value)
1: Fixed</td></tr>
<tr><td>20</td><td>SAD</td><td>0</td><td>R/W</td><td>Sets the source address counting direction of DMA channel n.
0: Increment (initial value)
1: Fixed</td></tr>
<tr><td>19 to 16</td><td>DDS[3:0]</td><td>0000</td><td>R/W</td><td>Destination Data Size
Sets the DMA transfer size of the transfer destination.
<table>
<tr><th>Value</th><th>Size</th><th>Remark</th></tr>
<tr><td>0000</td><td>8 bits</td><td>Initial value</td></tr>
<tr><td>0001</td><td>16 bits</td><td></td></tr>
<tr><td>0010</td><td>32 bits</td><td></td></tr>
<tr><td>0011</td><td>64 bits</td><td></td></tr>
<tr><td>0100</td><td>128 bits</td><td></td></tr>
<tr><td>0101</td><td>256 bits</td><td></td></tr>
<tr><td>0110</td><td>512 bits</td><td></td></tr>
<tr><td>0111</td><td>1024 bits</td><td></td></tr>
<tr><td>Other than the above</td><td>—</td><td>Setting prohibited</td></tr>
</table></td></tr>
<tr><td>15 to 12</td><td>SDS[3:0]</td><td>0000</td><td>R/W</td><td>Source Data Size
Sets the DMA transfer size of the transfer source.
<table>
<tr><th>Value</th><th>Size</th><th>Remark</th></tr>
<tr><td>0000</td><td>8 bits</td><td>Initial value</td></tr>
<tr><td>0001</td><td>16 bits</td><td></td></tr>
<tr><td>0010</td><td>32 bits</td><td></td></tr>
<tr><td>0011</td><td>64 bits</td><td></td></tr>
<tr><td>0100</td><td>128 bits</td><td></td></tr>
<tr><td>0101</td><td>256 bits</td><td></td></tr>
<tr><td>0110</td><td>512 bits</td><td></td></tr>
<tr><td>0111</td><td>1024 bits</td><td></td></tr>
<tr><td>Other than the above</td><td>—</td><td>Setting prohibited</td></tr>
</table></td></tr>
<tr><td>11</td><td>—</td><td>0</td><td>R</td><td>Reserved. Set 0. A read operation results in 0 being read.</td></tr>
<tr><td>10 to 8</td><td>AM[2:0]</td><td>000</td><td>R/W</td><td>ACK Mode
Sets the DMAACK output mode.
000: (initial value)
001: Level mode (active until the transfer request from an on-chip peripheral module input becomes inactive)
01x: Bus cycle mode (active while the DMA transfer is in a bus cycle)
1xx: DMAACK not to be output (this setting should be made when an auto request is made by STG (CHCTRL_n/nS) or when the SCIFA transfer is performed)</td></tr>
</table>

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved. Set 0. A read operation results in 0 being read.
6	LVL	0	R/W	Level Selects whether to detect a DMA request based on the level or edge of the signal. 0: Detects based on the edge (initial value). 1: Detects based on the level.
5	HIEN	0	R/W	High Enable Selects whether to detect a DMA request using the High level or rising edge of the signal. When LVL = 0: HIEN = 1: Detects a request in response to the rising edge of the signal. HIEN = 0: Does not detect a request in response to the rising edge of the signal (initial value). When LVL = 1: HIEN = 1: Detects a request when the signal is at the High level. HIEN = 0: Does not detect a request even when the signal is at the High level (initial value).
4	LOEN	0	R/W	Low Enable Selects whether to detect a DMA request using the Low level or falling edge of the signal. When LVL = 0: LOEN = 1: Detects a request in response to the falling edge of the signal. LOEN = 0: Does not detect a request in response to the falling edge of the signal (initial value). When LVL = 1: LOEN = 1: Detects a request when the signal is at the Low level. LOEN = 0: Does not detect a request even when the signal is at the Low level (initial value).
3	REQD	0	R/W	Request Direction Selects whether DMAREQ selected by the SEL bit is the source or destination. This bit is also used to define when DMAACK is to become active. 0: Source; DMAACK is to become active when read (initial value). 1: Destination; DMAACK is to become active when written.
2 to 0	SEL[2:0]	000	R/W	These bits are used to set a DMAC channel. Set one of the following values so that the channel set by the SEL bits matches the CHCFG_n/nS channel. 000: CH0/CH8 001: CH1/CH9 010: CH2/CH10 011: CH3/CH11 100: CH4/CH12 101: CH5/CH13 110: CH6/CH14 111: CH7/CH15

14.4.10 Channel Interval Register n/nS (CHITVL_n/nS)

This register sets the transfer interval for DMA channel n (n = 0 to 15).

For details, see **Section 14.7.6, Interval Count Function**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ITVL															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
15 to 0	ITVL	All 0	R/W	Sets the channel transfer interval.

14.4.11 Channel Extension Register n/nS (CHEXT_n/nS)

This is an extension register for DMA channel n (n = 0 to 15).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCA[3:0]				—	DPR[2:0]			SCA[3:0]				—	SPR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Set 0. A read operation results in 0 being read.
15 to 12	DCA[3:0]	0000	R/W	Destination CACHE Sets the value to be output to AWCACHE[3:0] for DMA write transfer. See Note 1 below.
11	—	0	R	Set 0. A read operation results in 0 being read.
10 to 8	DPR[2:0]	000	R/W	Destination PROT Sets the value to be output to AWPROT[2:0] for DMA write transfer. In the case of non-secure access, the value output as AWPROT[1] is fixed to 1 regardless of the setting of the DPR[1] bit. See Note 2 below.
7 to 4	SCA[3:0]	0000	R/W	Source CACHE Sets the value to be output to ARCACHE[3:0] for DMA read transfer. See Note 1 below.
3	—	0	R	Set 0. A read operation results in 0 being read.
2 to 0	SPR[2:0]	000	R/W	Source PROT Sets the value to be output to ARPROT[2:0] for DMA read transfer. In the case of non-secure access, the value output as ARPROT[1] is fixed to 1 regardless of the setting of the SPR[1] bit. See Note 2 below.

Note 1. Cache support: Bits SCA and DCA are used to change the settings.

Note 2. Protection unit support: Bits SPR and DPR are used to change the settings.
For the setting value, see AMBA AXI Protocol Specification from Arm Limited.

14.4.12 Next Link Address Register n/nS (NXLA_n/nS)

This is a 32-bit register that sets the link address of DMA channel n (n = 0 to 15).

For information about the link mode, see **Section 14.6.3, Link Mode**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NXLA	All 0	R/W	Sets a link address. The low-order 2 bits are masked with 0s. Only an address aligned with a 4-byte boundary can be set.

14.4.13 Current Link Address Register n/nS (CRLA_n/nS)

This is a 32-bit register that indicates the link address of DMA channel n (n = 0 to 15).

For information about the link mode, see **Section 14.6.3, Link Mode**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA	All 0	R	Indicates the address of the currently executed descriptor.

14.4.14 DMA Control Register (DCTRL_0_7/0_7S, DCTRL_8_15/8_15S)

This register sets the transfer type for descriptor access and the arbitration between channels.
(DCTRL_0_7/0_7S is common for channels 0 to 7 and DCTRL_8_15/8_15S is common for channels 8 to 15.)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LWCA				—	LWPR			LDCA				—	LDPR		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LVINT	PR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	LWCA	0000	R/W	Link WriteBack CACHE Sets the value to be output to AWCACHE[3:0] during descriptor writeback in link mode. For the setting value, see Note in **Section 14.4.11, Channel Extension Register n/nS (CHEXT_n/nS)**.
27	—	0	R	Reserved. Set 0. The initial value is 0.
26 to 24	LWPR	000	R/W	Link WriteBack PROT Sets the value to be output to AWPROT[2:0] during descriptor writeback in link mode. In the case of non-secure access, the value output as AWPROT[1] is fixed to 1 regardless of the setting of the LWPR[1] bit. For the setting value, see AMBA AXI Protocol Specification from Arm Limited.
23 to 20	LDCA	0000	R/W	Link Descriptor CACHE Sets the value to be output to ARCACHE[3:0] during descriptor load in link mode. For the setting value, see Note in **Section 14.4.11, Channel Extension Register n/nS (CHEXT_n/nS)**.
19	—	0	R	Reserved. Set 0. The initial value is 0.
18 to 16	LDPR	000	R/W	Link Descriptor PROT Sets the value to be output to ARPROT[2:0] during descriptor load in link mode. In the case of non-secure access, the value output as ARPROT[1] is fixed to 1 regardless of the setting of the LDPR[1] bit. For the setting value, see AMBA AXI Protocol Specification from Arm Limited.
15 to 2	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
1	LVINT	0	R/W	Sets whether to use pulse output or level output for the DMA transfer end interrupt and DMA error interrupt. Set pulse output for this product. 0: Pulse output (initial value) 1: Level output
0	PR	0	R/W	Sets the transfer priority control mode between channels (see **Section 14.7.2, Priority Control for DMA Channels**). 0: Fixed priority mode (initial value) 1: Round robin mode

14.4.15 DMA Status EN Register (DSTAT_EN_0_7/0_7S)

This register indicates the EN bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	EN7	0	R	Indicates the EN bit status of DMA channel 7.
6	EN6	0	R	Indicates the EN bit status of DMA channel 6.
5	EN5	0	R	Indicates the EN bit status of DMA channel 5.
4	EN4	0	R	Indicates the EN bit status of DMA channel 4.
3	EN3	0	R	Indicates the EN bit status of DMA channel 3.
2	EN2	0	R	Indicates the EN bit status of DMA channel 2.
1	EN1	0	R	Indicates the EN bit status of DMA channel 1.
0	EN0	0	R	Indicates the EN bit status of DMA channel 0.

14.4.16 DMA Status EN Register (DSTAT_EN_8_15/8_15S)

This register indicates the EN bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	EN15	0	R	Indicates the EN bit status of DMA channel 15.
6	EN14	0	R	Indicates the EN bit status of DMA channel 14.
5	EN13	0	R	Indicates the EN bit status of DMA channel 13.
4	EN12	0	R	Indicates the EN bit status of DMA channel 12.
3	EN11	0	R	Indicates the EN bit status of DMA channel 11.
2	EN10	0	R	Indicates the EN bit status of DMA channel 10.
1	EN9	0	R	Indicates the EN bit status of DMA channel 9.
0	EN8	0	R	Indicates the EN bit status of DMA channel 8.

14.4.17 DMA Status ER Register (DSTAT_ER_0_7/0_7S)

This register indicates the ER bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	ER7	0	R	Indicates the ER bit status of DMA channel 7.
6	ER6	0	R	Indicates the ER bit status of DMA channel 6.
5	ER5	0	R	Indicates the ER bit status of DMA channel 5.
4	ER4	0	R	Indicates the ER bit status of DMA channel 4.
3	ER3	0	R	Indicates the ER bit status of DMA channel 3.
2	ER2	0	R	Indicates the ER bit status of DMA channel 2.
1	ER1	0	R	Indicates the ER bit status of DMA channel 1.
0	ER0	0	R	Indicates the ER bit status of DMA channel 0.

14.4.18 DMA Status ER Register (DSTAT_ER_8_15/8_15S)

This register indicates the ER bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ER15	ER14	ER13	ER12	ER11	ER10	ER9	ER8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	ER15	0	R	Indicates the ER bit status of DMA channel 15.
6	ER14	0	R	Indicates the ER bit status of DMA channel 14.
5	ER13	0	R	Indicates the ER bit status of DMA channel 13.
4	ER12	0	R	Indicates the ER bit status of DMA channel 12.
3	ER11	0	R	Indicates the ER bit status of DMA channel 11.
2	ER10	0	R	Indicates the ER bit status of DMA channel 10.
1	ER9	0	R	Indicates the ER bit status of DMA channel 9.
0	ER8	0	R	Indicates the ER bit status of DMA channel 8.

14.4.19 DMA Status END Register (DSTAT_END_0_7/0_7S)

This register indicates the END bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	END7	END6	END5	END4	END3	END2	END1	END0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	END7	0	R	Indicates the END bit status of DMA channel 7.
6	END6	0	R	Indicates the END bit status of DMA channel 6.
5	END5	0	R	Indicates the END bit status of DMA channel 5.
4	END4	0	R	Indicates the END bit status of DMA channel 4.
3	END3	0	R	Indicates the END bit status of DMA channel 3.
2	END2	0	R	Indicates the END bit status of DMA channel 2.
1	END1	0	R	Indicates the END bit status of DMA channel 1.
0	END0	0	R	Indicates the END bit status of DMA channel 0.

14.4.20 DMA Status END Register (DSTAT_END_8_15/8_15S)

This register indicates the END bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	END15	END14	END13	END12	END11	END10	END9	END8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	END15	0	R	Indicates the END bit status of DMA channel 15.
6	END14	0	R	Indicates the END bit status of DMA channel 14.
5	END13	0	R	Indicates the END bit status of DMA channel 13.
4	END12	0	R	Indicates the END bit status of DMA channel 12.
3	END11	0	R	Indicates the END bit status of DMA channel 11.
2	END10	0	R	Indicates the END bit status of DMA channel 10.
1	END9	0	R	Indicates the END bit status of DMA channel 9.
0	END8	0	R	Indicates the END bit status of DMA channel 8.

14.4.21 DMA Status TC Register (DSTAT_TC_0_7/0_7S)

This register indicates the TC bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	TC7	0	R	Indicates the TC bit status of DMA channel 7.
6	TC6	0	R	Indicates the TC bit status of DMA channel 6.
5	TC5	0	R	Indicates the TC bit status of DMA channel 5.
4	TC4	0	R	Indicates the TC bit status of DMA channel 4.
3	TC3	0	R	Indicates the TC bit status of DMA channel 3.
2	TC2	0	R	Indicates the TC bit status of DMA channel 2.
1	TC1	0	R	Indicates the TC bit status of DMA channel 1.
0	TC0	0	R	Indicates the TC bit status of DMA channel 0.

14.4.22 DMA Status TC Register (DSTAT_TC_8_15/8_15S)

This register indicates the TC bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	TC15	0	R	Indicates the TC bit status of DMA channel 15.
6	TC14	0	R	Indicates the TC bit status of DMA channel 14.
5	TC13	0	R	Indicates the TC bit status of DMA channel 13.
4	TC12	0	R	Indicates the TC bit status of DMA channel 12.
3	TC11	0	R	Indicates the TC bit status of DMA channel 11.
2	TC10	0	R	Indicates the TC bit status of DMA channel 10.
1	TC9	0	R	Indicates the TC bit status of DMA channel 9.
0	TC8	0	R	Indicates the TC bit status of DMA channel 8.

14.4.23 DMA Status SUS Register (DSTAT_SUS_0_7/0_7S)

This register indicates the SUS bit status of the CHSTAT_n/nS register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SUS7	SUS6	SUS5	SUS4	SUS3	SUS2	SUS1	SUS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	SUS7	0	R	Indicates the SUS bit status of DMA channel 7.
6	SUS6	0	R	Indicates the SUS bit status of DMA channel 6.
5	SUS5	0	R	Indicates the SUS bit status of DMA channel 5.
4	SUS4	0	R	Indicates the SUS bit status of DMA channel 4.
3	SUS3	0	R	Indicates the SUS bit status of DMA channel 3.
2	SUS2	0	R	Indicates the SUS bit status of DMA channel 2.
1	SUS1	0	R	Indicates the SUS bit status of DMA channel 1.
0	SUS0	0	R	Indicates the SUS bit status of DMA channel 0.

14.4.24 DMA Status SUS Register (DSTAT_SUS_8_15/8_15S)

This register indicates the SUS bit status of the CHSTAT_n/nS register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SUS15	SUS14	SUS13	SUS12	SUS11	SUS10	SUS9	SUS8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved. Set 0. A read operation results in 0 being read.
7	SUS15	0	R	Indicates the SUS bit status of DMA channel 15.
6	SUS14	0	R	Indicates the SUS bit status of DMA channel 14.
5	SUS13	0	R	Indicates the SUS bit status of DMA channel 13.
4	SUS12	0	R	Indicates the SUS bit status of DMA channel 12.
3	SUS11	0	R	Indicates the SUS bit status of DMA channel 11.
2	SUS10	0	R	Indicates the SUS bit status of DMA channel 10.
1	SUS9	0	R	Indicates the SUS bit status of DMA channel 9.
0	SUS8	0	R	Indicates the SUS bit status of DMA channel 8.

14.4.25 DMA Extension Resource Selectors 0/0S to 7/7S (DMARS0/0S to DMARS7/7S)

DMARSn/nS (n = 0 to 7) are 32-bit readable/writable registers that specify the DMA transfer sources from peripheral modules in each channel. DMARS0/0S is for channels 0/0S and 1/1S, DMARS1/1S is for channels 2/2S and 3/3S, and so on.

Table 14.3 shows the specifiable combinations.

Some on-chip peripheral modules in this product use the same signal both for an interrupt request and for a DMA transfer request. If such a module is selected by a DMARSn/nS register, the signal works as a DMA transfer request signal and interrupt requests to the interrupt controller are masked. To enable the interrupt, clear the setting of DMARSn/nS (set all MID[7:0] and RID[1:0] to 0). See **Table 14.3** for possible combinations of MID[7:0] and RID[1:0].

- DMARS0/0S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH1 MID[7:0]								CH1 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH0 MID[7:0]								CH0 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS1/1S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH3 MID[7:0]								CH3 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH2 MID[7:0]								CH2 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS2/2S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH5 MID[7:0]								CH5 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH4 MID[7:0]								CH4 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS3/3S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH7 MID[7:0]								CH7 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH6 MID[7:0]								CH6 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS4/4S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH9 MID[7:0]								CH9 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH8 MID[7:0]								CH8 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS5/5S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH11 MID[7:0]								CH11 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH10 MID[7:0]								CH10 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS6/6S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH13 MID[7:0]								CH13 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH12 MID[7:0]								CH12 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS7/7S

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CH15 MID[7:0]								CH15 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CH14 MID[7:0]								CH14 RID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

14.5 Operation

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority order; when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in two modes: auto request, and on-chip peripheral module request.

14.5.1 Transfer Flow

After the next source address register (N0SA_n/nS, N1SA_n/nS), next destination address register (N0DA_n/nS, N1DA_n/nS), next transaction byte register (N0TB_n/nS, N1TB_n/nS), channel control register (CHCTRL_n/nS), channel configuration register (CHCFG_n/nS), channel extension register (CHEXT_n/nS), DMA control register (DCTRL_0_7/0_7S, DCTRL_8_15/8_15S), and DMA extension resource selector (DMARSn/nS) are set for the target transfer conditions, the DMAC transfers data according to the following procedure:

1. Checks to see if transfer is enabled (EN = 0 and TACT = 0 in channel status register).
2. Clears the channel status register (set 1 in the SWRST bit of the channel control register).
3. Enables DMA transfer (set 1 in the SETEN bit of the channel control register).
4. When a transfer request comes and transfer is enabled, the DMAC transfers one transfer unit of data (depending on the DDS[3:0] and SDS[3:0] bit settings). For an auto request, the transfer begins automatically when 1 is set in the STG bit of the channel control register. The CRTB_n/nS value will be decremented by 1 for each transfer.
5. If 0 is set in the REN bit of the channel configuration register when transfer has been completed for the specified count (when CRTB_n/nS reaches 0), transfer ends normally. If the DEM bit of the channel configuration register is set to 0 at this time, a DMA transfer end interrupt is sent to the CPU. If the REN bit is 1 when CRTB_n/nS reaches 0, transfer operations are continued with the values of N0SA_n/nS, N1SA_n/nS, N0DA_n/nS, N1DA_n/nS, N0TB_n/nS, and N1TB_n/nS set by the RSEL bit of the channel configuration register until there are no more transfer requests.
6. When an address error in the DMAC is generated, the transfer is stopped. Transfers are also stopped when 1 is set in the CLREN bit of CHCTRL_n/nS.

14.5.2 DMA Transfer Requests

DMA transfer requests are basically generated in either the data transfer source or destination, but they can also be generated in on-chip peripheral modules that are neither the transfer source nor destination. Transfers can be requested in two modes: auto request, and on-chip peripheral module request. On-chip peripheral module request is selected by the DMARS0/0S to DMARS7/7S registers.

(1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the DMAC to automatically generate a transfer request signal internally. When the STG bit in channel control register n is set to 1, the transfer begins so long as the TACT bit in channel status register is 0.

(2) On-Chip Peripheral Module Request Mode

In this mode, the transfer is performed in response to the DMA transfer request signal from an on-chip peripheral module.

When a transfer request signal is sent in on-chip peripheral module request mode while DMA transfer is enabled, the DMA transfer is performed.

The DMA transfer request signals to be sent from on-chip peripheral modules or external pin input are listed in

Table 14.3.

The transfer source or destination is fixed for some on-chip peripheral module requests. For details, see **Table 14.3**.

Table 14.3 On-Chip Module Requests (1/8)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS[*1]						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GTM ch0	OSTM0TINT (compare match)	Arbitrary	Arbitrary	0000_1000	11	0/1	010	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GTM ch1	OSTM1TINT (compare match)	Arbitrary	Arbitrary	0000_1001	11	0/1	010	0	1	0	0/1	
GTM ch2	OSTM2TINT (compare match)	Arbitrary	Arbitrary	0000_1010	11	0/1	010	0	1	0	0/1	
MTU3 ch0	TGIA0 (input capture/compare match)	Arbitrary	Arbitrary	0000_1100	11	0/1	001	0	1	0	0/1	
	TGIB0 (input capture/compare match)	Arbitrary	Arbitrary	0000_1101	11	0/1	001	0	1	0	0/1	
	TGIC0 (input capture/compare match)	Arbitrary	Arbitrary	0000_1110	11	0/1	001	0	1	0	0/1	
	TGID0 (input capture/compare match)	Arbitrary	Arbitrary	0000_1111	11	0/1	001	0	1	0	0/1	
MTU3 ch1	TGIA1 (input capture/compare match)	Arbitrary	Arbitrary	0001_0000	11	0/1	001	0	1	0	0/1	
	TGIB1 (input capture/compare match)	Arbitrary	Arbitrary	0001_0001	11	0/1	001	0	1	0	0/1	
MTU3 ch2	TGIA2 (input capture/compare match)	Arbitrary	Arbitrary	0001_0010	11	0/1	001	0	1	0	0/1	
	TGIB2 (input capture/compare match)	Arbitrary	Arbitrary	0001_0011	11	0/1	001	0	1	0	0/1	
MTU3 ch3	TGIA3 (input capture/compare match)	Arbitrary	Arbitrary	0001_0100	11	0/1	001	0	1	0	0/1	
	TGIB3 (input capture/compare match)	Arbitrary	Arbitrary	0001_0101	11	0/1	001	0	1	0	0/1	
	TGIC3 (input capture/compare match)	Arbitrary	Arbitrary	0001_0110	11	0/1	001	0	1	0	0/1	
	TGID3 (input capture/compare match)	Arbitrary	Arbitrary	0001_0111	11	0/1	001	0	1	0	0/1	
MTU3 ch4	TGIA4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1000	11	0/1	001	0	1	0	0/1	
	TGIB4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1001	11	0/1	001	0	1	0	0/1	
	TGIC4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1010	11	0/1	001	0	1	0	0/1	
	TGID4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1011	11	0/1	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (2/8)

DMA Transfer

				DMARSn/nS		CHCFG_n/nS*1						
Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
MTU3 ch4	TCIV4 (input capture/compare match)	Arbitrary	Arbitrary	0001_1100	11	0/1	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
MTU3 ch5	TGIU5 (input capture/compare match)	Arbitrary	Arbitrary	0001_1101	11	0/1	001	0	1	0	0/1	
	TGIV5 (input capture/compare match)	Arbitrary	Arbitrary	0001_1110	11	0/1	001	0	1	0	0/1	
	TGIW5 (input capture/compare match)	Arbitrary	Arbitrary	0001_1111	11	0/1	001	0	1	0	0/1	
MTU3 ch6	TGIA6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0000	11	0/1	001	0	1	0	0/1	
	TGIB6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0001	11	0/1	001	0	1	0	0/1	
	TGIC6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0010	11	0/1	001	0	1	0	0/1	
	TGID6 (input capture/compare match)	Arbitrary	Arbitrary	0010_0011	11	0/1	001	0	1	0	0/1	
MTU3 ch7	TGIA7 (input capture/compare match)	Arbitrary	Arbitrary	0010_0100	11	0/1	001	0	1	0	0/1	
	TGIB7 (input capture/compare match)	Arbitrary	Arbitrary	0010_0101	11	0/1	001	0	1	0	0/1	
	TGIC7 (input capture/compare match)	Arbitrary	Arbitrary	0010_0110	11	0/1	001	0	1	0	0/1	
	TGID7 (input capture/compare match)	Arbitrary	Arbitrary	0010_0111	11	0/1	001	0	1	0	0/1	
	TCIV7 (input capture/compare match)	Arbitrary	Arbitrary	0010_1000	11	0/1	001	0	1	0	0/1	
MTU3 ch8	TGIA8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1001	11	0/1	001	0	1	0	0/1	
	TGIB8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1010	11	0/1	001	0	1	0	0/1	
	TGIC8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1011	11	0/1	001	0	1	0	0/1	
	TGID8 (input capture/compare match)	Arbitrary	Arbitrary	0010_1100	11	0/1	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (3/8)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
PWM(GPT) ch0	CCMPA0 (input capture/compare match)	Arbitrary	Arbitrary	0010_1101	11	0/1	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	CCMPB0 (input capture/compare match)	Arbitrary	Arbitrary	0010_1110	11	0/1	001	0	1	0	0/1	
	CMPC0 (compare match)	Arbitrary	Arbitrary	0010_1111	11	0/1	001	0	1	0	0/1	
	CMPD0 (compare match)	Arbitrary	Arbitrary	0011_0000	11	0/1	001	0	1	0	0/1	
	CMPE0 (compare match)	Arbitrary	Arbitrary	0011 0001	11	0	001	0	1	0	0/1	
	CMPF0 (compare match)	Arbitrary	Arbitrary	0011 0010	11	0	001	0	1	0	0/1	
	ADTRGA0 (compare match)	Arbitrary	Arbitrary	0011 0011	11	0	001	0	1	0	0/1	
	ADTRGA0 (compare match)	Arbitrary	Arbitrary	0011 0011	11	0/1	001	0	1	0	0/1	
	ADTRGB0 (compare match)	Arbitrary	Arbitrary	0011 0100	11	0/1	001	0	1	0	0/1	
	OVF0 (overflow)	Arbitrary	Arbitrary	0011 0101	11	0/1	001	0	1	0	0/1	
	UNF0 (underflow)	Arbitrary	Arbitrary	0011 0110	11	0/1	001	0	1	0	0/1	
PWM(GPT) ch1	CCMPA1 (input capture/compare match）	Arbitrary	Arbitrary	00111010	11	0/1	001	0	1	0	0/1	
	CCMPB1 (input capture/compare match）	Arbitrary	Arbitrary	00111011	11	0/1	001	0	1	0	0/1	
	CMPC1 (compare match)	Arbitrary	Arbitrary	00111100	11	0/1	001	0	1	0	0/1	
	CMPD1 (compare match)	Arbitrary	Arbitrary	00111101	11	0/1	001	0	1	0	0/1	
	CMPE1 (compare match)	Arbitrary	Arbitrary	00111110	11	0/1	001	0	1	0	0/1	
	CMPF1 (compare match)	Arbitrary	Arbitrary	00111111	11	0/1	001	0	1	0	0/1	
	ADTRGA1 (compare match)	Arbitrary	Arbitrary	01000000	11	0/1	001	0	1	0	0/1	
	ADTRGB1 (compare match)	Arbitrary	Arbitrary	01000001	11	0/1	001	0	1	0	0/1	
	OVF1 (overflow)	Arbitrary	Arbitrary	01000010	11	0/1	001	0	1	0	0/1	
	UNF1 (underflow)	Arbitrary	Arbitrary	01000011	11	0/1	001	0	1	0	0/1	
PWM(GPT) ch2	CCMPA2 (input capture/compare match）	Arbitrary	Arbitrary	01000111	11	0/1	001	0	1	0	0/1	
	CCMPB2 (input capture/compare match）	Arbitrary	Arbitrary	01001000	11	0/1	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (4/8)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
PWM(GPT) ch2	CMPC2 (compare match)	Arbitrary	Arbitrary	01001001	11	0/1	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	CMPD2 (compare match)	Arbitrary	Arbitrary	01001010	11	0	001	0	1	0	0/1	
	CMPE2 (compare match)	Arbitrary	Arbitrary	01001011	11	0	001	0	1	0	0/1	
	CMPF2 (compare match)	Arbitrary	Arbitrary	01001100	11	0	001	0	1	0	0/1	
	ADTRGA2 (compare match)	Arbitrary	Arbitrary	01001101	11	0/1	001	0	1	0	0/1	
	ADTRGB2 (compare match)	Arbitrary	Arbitrary	01001110	11	0/1	001	0	1	0	0/1	
	OVF2 (overflow)	Arbitrary	Arbitrary	01001111	11	0/1	001	0	1	0	0/1	
	UNF2 (underflow)	Arbitrary	Arbitrary	01010000	11	0/1	001	0	1	0	0/1	
PWM(GPT) ch3	CCMPA3 (input capture/compare match）	Arbitrary	Arbitrary	01010100	11	0/1	001	0	1	0	0/1	
	CCMPB3 (input capture/compare match）	Arbitrary	Arbitrary	01010101	11	0/1	001	0	1	0	0/1	
	CMPC3 (compare match)	Arbitrary	Arbitrary	01010110	11	0/1	001	0	1	0	0/1	
	CMPD3 (compare match)	Arbitrary	Arbitrary	01010111	11	0/1	001	0	1	0	0/1	
	CMPE3 (compare match)	Arbitrary	Arbitrary	01011000	11	0/1	001	0	1	0	0/1	
	CMPF3 (compare match)	Arbitrary	Arbitrary	01011001	11	0/1	001	0	1	0	0/1	
	ADTRGA3 (compare match)	Arbitrary	Arbitrary	01011010	11	0/1	001	0	1	0	0/1	
	ADTRGB3 (compare match)	Arbitrary	Arbitrary	01011011	11	0/1	001	0	1	0	0/1	
	OVF3 (overflow)	Arbitrary	Arbitrary	01011100	11	0/1	001	0	1	0	0/1	
	UNF3 (underflow)	Arbitrary	Arbitrary	01011101	11	0/1	001	0	1	0	0/1	
PWM(GPT) ch4	CCMPA4 (input capture/compare match)	Arbitrary	Arbitrary	01100001	11	0/1	001	0	1	0	0/1	
	CCMPB4 (input capture/compare match)	Arbitrary	Arbitrary	01100010	11	0/1	001	0	1	0	0/1	
	CMPC4 (compare match)	Arbitrary	Arbitrary	01100011	11	0/1	001	0	1	0	0/1	
	CMPD4 (compare match)	Arbitrary	Arbitrary	01100100	11	0	001	0	1	0	0/1	
	CMPE4 (compare match)	Arbitrary	Arbitrary	01100101	11	0	001	0	1	0	0/1	
	CMPF4 (compare match)	Arbitrary	Arbitrary	01100110	11	0	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (5/8)

DMA Transfer

				DMARSn/nS		CHCFG_n/nS*1						
Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
PWM(GPT) ch4	ADTRGA4 (compare match)	Arbitrary	Arbitrary	01100111	11	0/1	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	ADTRGB4 (compare match)	Arbitrary	Arbitrary	01101000	11	0/1	001	0	1	0	0/1	
	OVF4 (overflow)	Arbitrary	Arbitrary	01101001	11	0/1	001	0	1	0	0/1	
	UNF4 (underflow)	Arbitrary	Arbitrary	01101010	11	0/1	001	0	1	0	0/1	
PWM(GPT) ch5	CCMPA5 (input capture/compare match)	Arbitrary	Arbitrary	01101110	11	0/1	001	0	1	0	0/1	
	CCMPB5 (input capture/compare match)	Arbitrary	Arbitrary	01101111	11	0/1	001	0	1	0	0/1	
	CMPC5 (compare match)	Arbitrary	Arbitrary	01110000	11	0/1	001	0	1	0	0/1	
	CMPD5 (compare match)	Arbitrary	Arbitrary	01110001	11	0/1	001	0	1	0	0/1	
	CMPE5 (compare match)	Arbitrary	Arbitrary	01110010	11	0/1	001	0	1	0	0/1	
	CMPF5 (compare match)	Arbitrary	Arbitrary	01110011	11	0/1	001	0	1	0	0/1	
	ADTRGA5 (compare match)	Arbitrary	Arbitrary	01110100	11	0/1	001	0	1	0	0/1	
	ADTRGB5 (compare match)	Arbitrary	Arbitrary	01110101	11	0/1	001	0	1	0	0/1	
	OVF5 (overflow)	Arbitrary	Arbitrary	01110110	11	0/1	001	0	1	0	0/1	
	UNF5 (underflow)	Arbitrary	Arbitrary	01110111	11	0/1	001	0	1	0	0/1	
PWM(GPT) ch6	CCMPA6 (input capture/compare match）	Arbitrary	Arbitrary	01111011	11	0/1	001	0	1	0	0/1	
	CCMPB6 (input capture/compare match)	Arbitrary	Arbitrary	01111100	11	0/1	001	0	1	0	0/1	
	CMPC6 (compare match)	Arbitrary	Arbitrary	01111101	11	0/1	001	0	1	0	0/1	
	CMPD6 (compare match)	Arbitrary	Arbitrary	01111110	11	0	001	0	1	0	0/1	
	CMPE6 (compare match)	Arbitrary	Arbitrary	01111111	11	0	001	0	1	0	0/1	
	CMPF6 (compare match)	Arbitrary	Arbitrary	10000000	11	0	001	0	1	0	0/1	
	ADTRGA6 (compare match)	Arbitrary	Arbitrary	10000001	11	0/1	001	0	1	0	0/1	
	ADTRGB6 (compare match)	Arbitrary	Arbitrary	10000010	11	0/1	001	0	1	0	0/1	
	OVF6 (overflow)	Arbitrary	Arbitrary	10000011	11	0/1	001	0	1	0	0/1	
	UNF6 (underflow)	Arbitrary	Arbitrary	10000100	11	0/1	001	0	1	0	0/1	

Table 14.3 On-Chip Module Requests (6/8)

DMA Transfer

				DMARSn/nS		CHCFG_n/nS*1						
Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
PWM(GPT) ch7	CCMPA7 (input capture/compare match)	Arbitrary	Arbitrary	10001000	11	0/1	001	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	CCMPB7 (input capture/compare match)	Arbitrary	Arbitrary	10001001	11	0/1	001	0	1	0	0/1	
	CMPC7 (compare match)	Arbitrary	Arbitrary	10001010	11	0/1	001	0	1	0	0/1	
	CMPD7 (compare match)	Arbitrary	Arbitrary	10001011	11	0/1	001	0	1	0	0/1	
	CMPE7 (compare match)	Arbitrary	Arbitrary	10001100	11	0/1	001	0	1	0	0/1	
	CMPF7 (compare match)	Arbitrary	Arbitrary	10001101	11	0/1	001	0	1	0	0/1	
	ADTRGA7 (compare match)	Arbitrary	Arbitrary	10001110	11	0/1	001	0	1	0	0/1	
	ADTRGB7 (compare match)	Arbitrary	Arbitrary	10001111	11	0/1	001	0	1	0	0/1	
	OVF7 (overflow)	Arbitrary	Arbitrary	10010000	11	0/1	001	0	1	0	0/1	
	UNF7 (underflow)	Arbitrary	Arbitrary	10010001	11	0/1	001	0	1	0	0/1	
SSIF ch0	INT_ssif_dma_rx_0 (receive data full)	SSIF RDR_0	Arbitrary	10010101	10	0	010	0	1	0	0	
	INT_ssif_dma_tx_0 (transmit data empty)	Arbitrary	SSIF TDR_0	10010101	01	0	010	0	1	0	1	
SSIF ch1	NT_ssif_dma_rx_1 (receive data full)	SSIF RDR_1	Arbitrary	10010110	10	0	010	0	1	0	0	
	INT_ssif_dma_tx_1 (transmit data empty)	Arbitrary	SSIF TDR_1	10010110	01	0	010	0	1	0	1	
SSIF ch2	INT_ssif_dma_rt_2 (receive data full)	SSIF RDR_2	Arbitrary	10010111	11	0	010	0	1	0	0	
	INT_ssif_dma_rt_2 (transmit data empty)	Arbitrary	SSIF TDR_2	10010111	11	0	010	0	1	0	1	
SSIF ch3	INT_ssif_dma_rx_3 (receive data full)	SSIF RDR_3	Arbitrary	10011000	10	0	010	0	1	0	0	
	INT_ssif_dma_tx_3 (transmit data empty)	Arbitrary	SSIF TDR_3	10011000	01	0	010	0	1	0	1	
SRC	SRC_IDEI (input data FIFO empty)	Arbitrary	SRCID	10011001	10	0	010	0	1	0	1	
	SRC_ODFI (output data FIFO full)	SRCOD	Arbitrary	10011001	01	0	010	0	1	0	0	
I2C ch0	INTRIIC_RI0 (receive data full)	RIIC0DRR	Arbitrary	10011010	10	0	010	0	1	0	0	
	INTRIIC_TI0 (transmit data empty)	Arbitrary	RIIC0DRT	10011010	01	0	010	0	1	0	1	
I2C ch1	INTRIIC_RI1 (receive data full)	RIIC1DRR	Arbitrary	10011011	10	0	010	0	1	0	0	
	INTRIIC_TI1 (transmit data empty)	Arbitrary	RIIC1DRT	10011011	01	0	010	0	1	0	1	

Table 14.3 On-Chip Module Requests (7/8)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS*1						
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
I2C ch2	INTRIIC_RI2 (receive data full)	RIIC2DRR	Arbitrary	10011100	10	0	010	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	INTRIIC_TI2 (transmit data empty)	Arbitrary	RIIC2DRT	10011100	01	0	010	0	1	0	1	
I2C ch3	INTRIIC_RI3 (receive data full)	RIIC3DRR	Arbitrary	10011101	10	0	010	0	1	0	0	
	INTRIIC_TI3 (transmit data empty)	Arbitrary	RIIC3DRT	10011101	01	0	010	0	1	0	1	
SCIF ch0	RXI0 (receive FIFO data full)	FRDR0	Arbitrary	10011110	10	0	100	1	1	0	0	
	TXI0 (transmit FIFO data empty)	Arbitrary	FTDR0	10011110	01	0	100	1	1	0	1	
SCIF ch1	RXI1 (receive FIFO data full)	FRDR1	Arbitrary	10011111	10	0	100	1	1	0	0	
	TXI1 (transmit FIFO data empty)	Arbitrary	FTDR1	10011111	01	0	100	1	1	0	1	
SCIF ch2	RXI2 (receive FIFO data full)	FRDR2	Arbitrary	10100000	10	0	100	1	1	0	0	
	TXI2 (transmit FIFO data empty)	Arbitrary	FTDR2	10100000	01	0	100	1	1	0	1	
SCIF ch3	RXI3 (receive FIFO data full)	FRDR3	Arbitrary	10100001	10	0	100	1	1	0	0	
	TXI3 (transmit FIFO data empty)	Arbitrary	FTDR3	10100001	01	0	100	1	1	0	1	
SCIF ch4	RXI4 (receive FIFO data full)	FRDR4	Arbitrary	10100010	10	0	100	1	1	0	0	
	TXI4 (transmit FIFO data empty)	Arbitrary	FTDR4	10100010	01	0	100	1	1	0	1	
SCIg ch0	RXI0 (receive FIFO data full)	RDR0	Arbitrary	10100011	10	0	010	1	1	0	0	
	TXI0 (transmit FIFO data empty)	Arbitrary	TDR0	10100011	01	0	010	1	1	0	1	
SCIg ch1	RXI1 (receive FIFO data full)	RDR1	Arbitrary	10100100	10	0	010	1	1	0	0	
	TXI1 (transmit FIFO data empty)	Arbitrary	TDR1	10100100	01	0	010	1	1	0	1	
RSPI ch0	SPRI0 (receive buffer full)	SPDR0	Arbitrary	10100101	10	0	010	1	1	0	0	
	SPTI0 (transmit buffer empty)	Arbitrary	SPDR0	10100101	01	0	010	1	1	0	1	
RSPI ch1	SPRI1 (receive buffer full)	SPDR1	Arbitrary	10100110	10	0	010	1	1	0	0	
	SPTI1 (transmit buffer empty)	Arbitrary	SPDR1	10100110	01	0	010	1	1	0	1	

Table 14.3 On-Chip Module Requests (8/8)

DMA Transfer

Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARSn/nS		CHCFG_n/nS[*1]							
				MID [7:0]	RID [1:0]	TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
RSPI ch2	SPRI2 (receive buffer full)	SPDR2	Arbitrary	10100111	10	0	010	1	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
	SPTI2 (transmit buffer empty)	Arbitrary	SPDR2	10100111	01	0	010	1	1	0	1	
TSIP	WRRDY1	Arbitrary	TSIP_REG	10101000	11	0	001	0	1	0	0/1	
	WRRDY0	Arbitrary	TSIP_REG	10101001	11	0	001	0	1	0	0/1	
	WRRDY4	Arbitrary	TSIP_REG	10101010	11	0	001	0	1	0	0/1	
	RDRDY1	TSIP_REG	Arbitrary	10101011	11	0	001	0	1	0	0/1	
	RDRDY0	TSIP_REG	Arbitrary	10101100	11	0	001	0	1	0	0/1	
	IRDRDY	TSIP_REG	Arbitrary	10101101	11	0	001	0	1	0	0/1	
	IWRRDY	Arbitrary	TSIP_REG	10101110	11	0	001	0	1	0	0/1	
CANFD	RXF_DMA0	RSCFD0CFDRFDF0_0	Arbitrary	10101111	11	0	001	0	1	0	0/1	
	RXF_DMA1	RSCFD0CFDRFDF0_1	Arbitrary	10110000	11	0	001	0	1	0	0/1	
	RXF_DMA2	RSCFD0CFDRFDF0_2	Arbitrary	10110001	11	0	001	0	1	0	0/1	
	RXF_DMA3	RSCFD0CFDRFDF0_3	Arbitrary	10110010	11	0	001	0	1	0	0/1	
	RXF_DMA4	RSCFD0CFDRFDF0_4	Arbitrary	10110011	11	0	001	0	1	0	0/1	
	RXF_DMA5	RSCFD0CFDRFDF0_5	Arbitrary	10110100	11	0	001	0	1	0	0/1	
	RXF_DMA6	RSCFD0CFDRFDF0_6	Arbitrary	10110101	11	0	001	0	1	0	0/1	
	RXF_DMA7	RSCFD0CFDRFDF0_7	Arbitrary	10110110	11	0	001	0	1	0	0/1	
	COM_DMA0 (transceiver FIFO receive mode only)	RSCFD0CFDRFDF0_0	Arbitrary	10110111	11	0	001	0	1	0	0/1	
	COM_DMA1 (transceiver FIFO Receive mode only)	RSCFD0CFDRFDF0_1	Arbitrary	10111000	11	0	001	0	1	0	0/1	

Note 1. CHCFG_n/nS setting value

TM	0: Single transfer 1: Block transfer
AM	001: ACK level output 010: ACK bus cycle output 100: No ACK
LVL	0: REQ edge detection 1: REQ level detection
REQD	0: ACK output at read 1: ACK output at write

Note 2. Only in products with a Trusted Secure IP

14.6 DMA Mode

14.6.1 Mode Setting

The DMS field of the CHCFG_n/nS register can be used to toggle between register mode and link mode.

Table 14.4 DMA Mode Setting

DMS (CHCFG_n/nS)	Mode	Description
0	Register mode	A DMA transfer is executed using the values set in the Next Register Set.
1	Link mode	A DMA transfer is executed using the descriptor set in the Current register. The DMAC repeatedly loads the descriptor and executes the DMA transfer unless otherwise set by the descriptor or stopped by the control register.

14.6.2 Register Mode

In register mode, a DMA transfer is executed using the values set in the internal registers.

Two sets of the source address, destination address, and transfer byte count (Next0 Register Set and Next1 Register Set) can be set. It is possible to select the Next register set to be used for the DMA transfer, as well as to execute two Next register sets continuously for the DMA transfer.

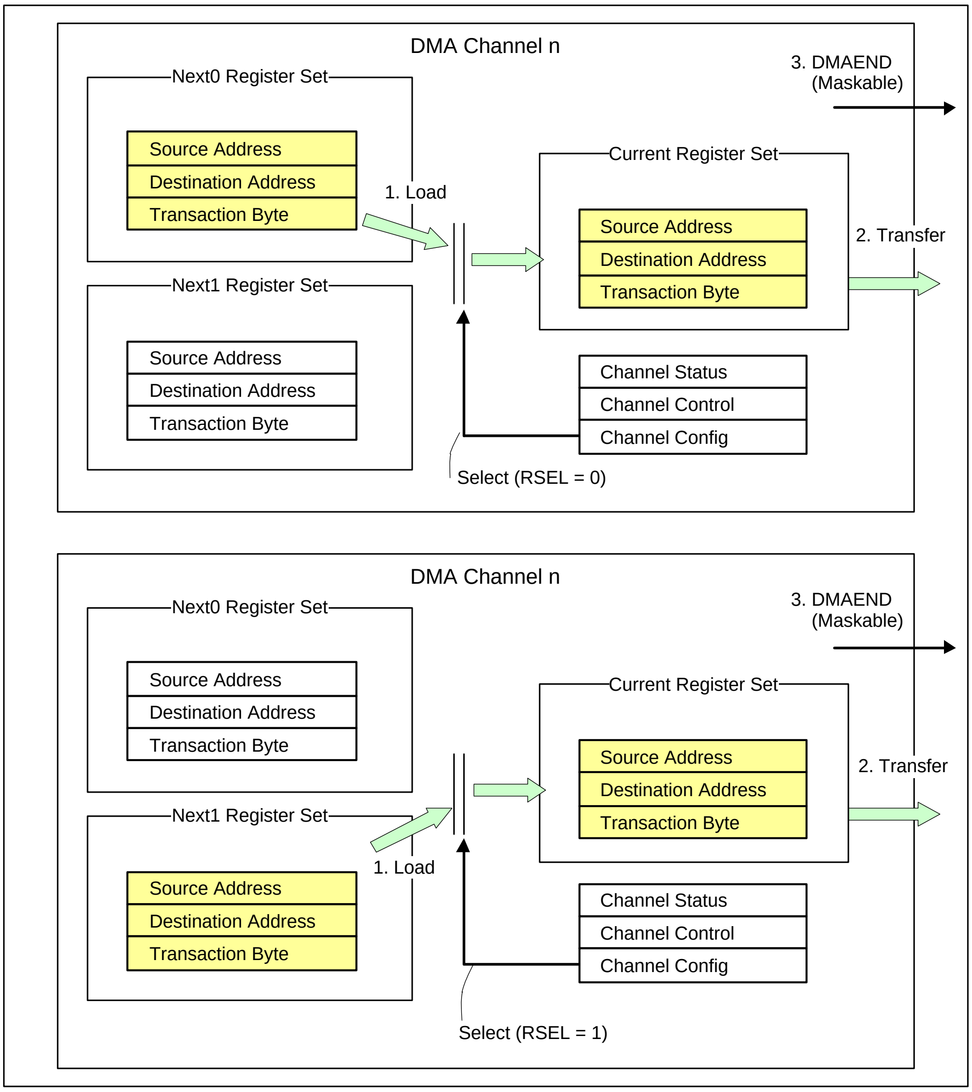


Figure 14.2 Outline of Normal Register Mode

The above figure shows how the transfer is executed when the Next0 Register Set is used (upper part of the figure) and when the Next1 Register Set is used (lower part of the figure).

(1) Operation Flow

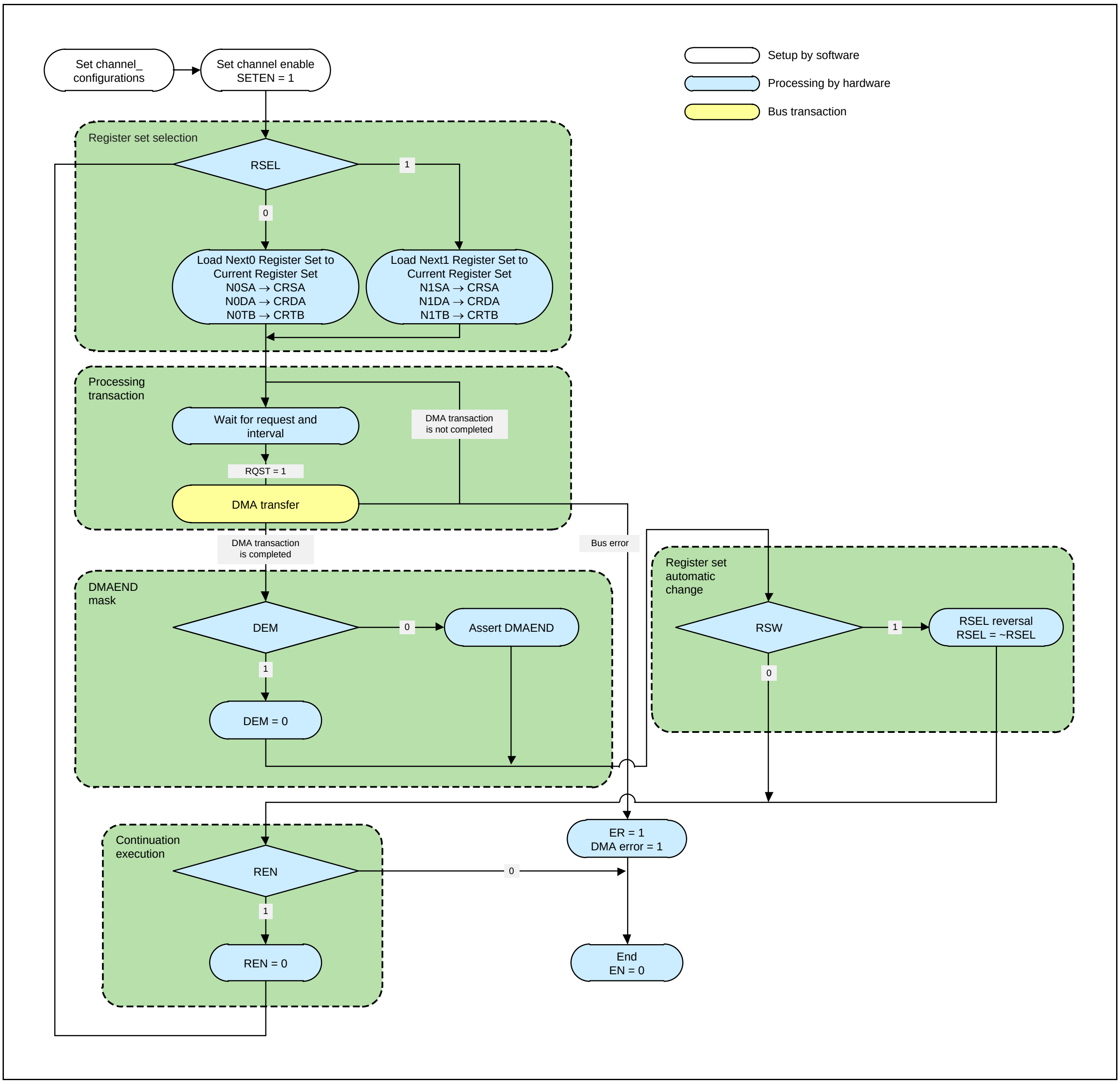


Figure 14.3 Register Mode Flow

<Explanation of the register mode flow>

1. Channel setting (set channel configuration)

   The Next0 or Next1 register set (destination address, source address, and total transfer byte count) is set.

   In the Channel register set, the DMA register set (REQ, DMAACK, transfer size, etc.) is set. (See **Section 14.7, DMA Transfer**.)

2. Register set selection (register set selection)

   When 1 is set in EN, the values set in the Next register set selected by RSEL are loaded to the Current register set.

3. DMA transaction (processing transaction)

   A DMA transfer is executed according to the set values. For details of the transfer, see **Section 14.7, DMA Transfer**.

4. DMA transfer end interrupt mask (DMAINT mask)

   The DMA transfer end interrupt is masked according to the value set in the DEM bit of CHCFG_n/nS. When 1 is set in DEM, the DMA transfer end interrupt is not output. Also, immediately after that, DEM is automatically cleared to 0.

5. Automatic register set change (register set automatic change)

   Whether to use the other Next register set is determined by the value set in the RSW bit of CHCFG_n/nS.

6. Continuation of execution (continuation execution)

   Whether to continue the execution of the DMA transfer is determined by the value set in the REN bit of CHCFG_n/ nS. When 1 is set in REN, the execution of the DMA transfer is continued. Also, immediately after that, REN is automatically cleared to 0.

(2) Register Setting

(a) Register mode setting

Select the register set to be executed.

Table 14.5 Register Mode Setting

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	Description
0	0	Executes the Next0 Register Set.
	1	Executes the Next1 Register Set.

(b) DMA transfer end interrupt mask setting

The DMA transfer end interrupt can be masked individually for each register set.

Table 14.6 DMAINT Mask Setting

DEM (CHCFG_n/nS)	Operation
0	When the DMA transaction is completed, a DMA transfer end interrupt is issued.
1	Even when the DMA transaction is completed, a DMA transfer end interrupt is not issued. After the DMA transaction is completed, DEM is cleared to 0 by hardware.

(c) Automatic register set execution setting

After DMA transfers, the DMA transaction of the selected register set is automatically executed.

Table 14.7 Automatic Register Set Execution Setting

REN (CHCFG_n/nS)	Operation	Remark
0	When the DMA transaction of the register set selected by RSEL is completed, the EN bit is cleared and the DMA operation ends.	Set this value to execute a DMA transaction once.
1	When a DMA transaction is completed, the DMAC continues to execute a DMA transfer by using the data set in the selected register set. When continuous transfers are successful, REN is cleared to 0.	Set this value to continuously execute DMA transfers by using the data set in separate register sets.

(d) Automatic register set change setting

When 1 is set in REN, the DMAC can automatically change to the register set to be executed next, after a DMA transaction is completed.

Table 14.8 Automatic Register Set Change Setting

RSW (CHCFG_n/nS)	Operation	Remark
0	If 1 is set in REN when a DMA transaction is completed, the register set is not changed.	Set this value to use only one register set.
1	If 1 is set in REN when a DMA transaction is completed, the value of RSEL is automatically inverted and the other register set is selected.	Set this value to change the register set.

(3) Setting Examples

(a) When only the Next0 register set is used

Table 14.9 Register Mode Setting Example 1

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	DEM (CHCFG_n/nS)	RSW (CHCFG_n/nS)	REN (CHCFG_n/nS)
0 (Register mode)	0 (Next0)	0 (not masked)	0 (not switched)	0 (not continuously executed)

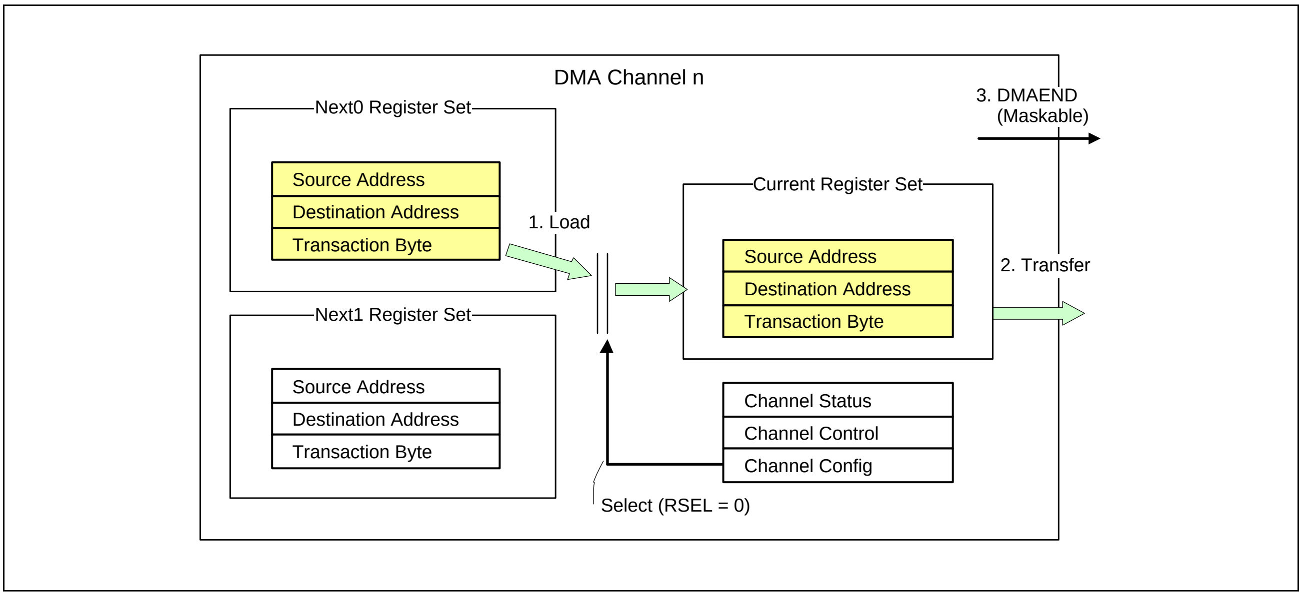


Figure 14.4 Register Mode Setting Example 1

1. 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
2. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
3. Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
4. Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.

(b) When two register sets are used continuously

Table 14.10 Automatic Register Set Execution Setting

DMS (CHCFG_n/nS)	RSEL (CHCFG_n/nS)	DEM (CHCFG_n/nS)	RSW (CHCFG_n/nS)	REN (CHCFG_n/nS)
0 (Register mode)	0 (Next0)	1 (masked)	1 (switched)	1 (continuously executed)

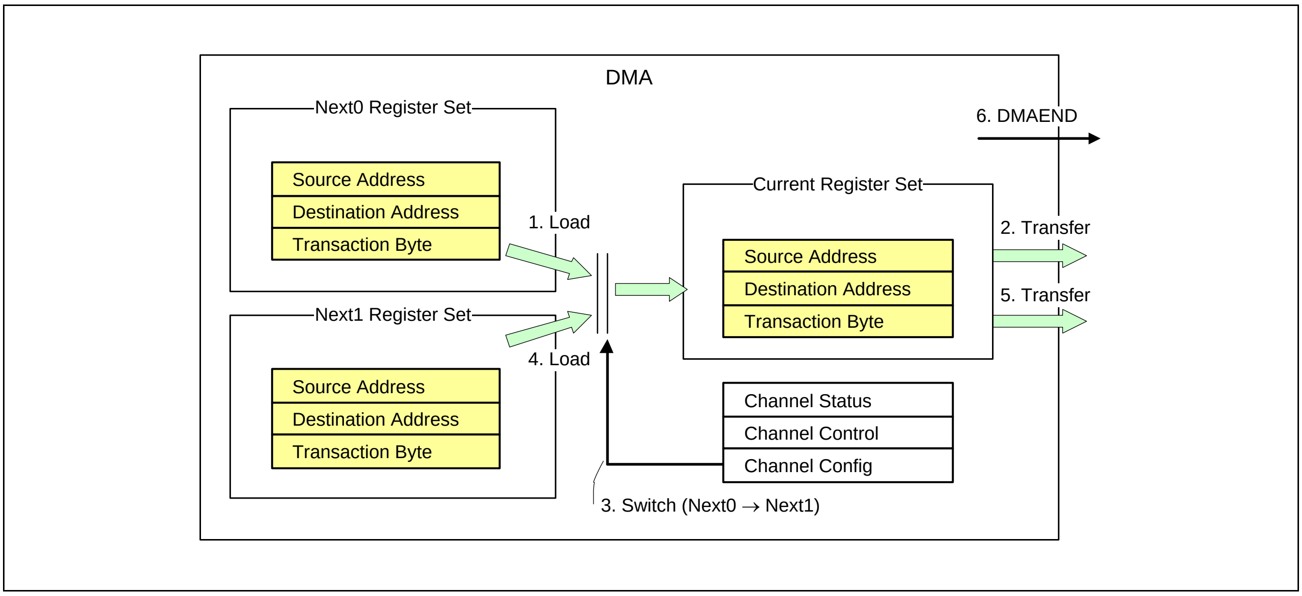


Figure 14.5 Register Mode Setting Example 2

1. 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
2. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
3. Because 1 is set in DEM, DMA transfer end interrupt is not output after the DMA transaction is completed. Also, DEM is automatically cleared to 0.
4. Because 1 is set in REN, the execution is continued. Also, REN is automatically cleared to 0.
5. Because 1 is set in RSW, the register set to be executed next is switched (RSEL = 0 → 1).
6. The Next1 register set is loaded to the Current register set.
7. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
8. Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
9. Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.

14.6.3 Link Mode

In link mode, a descriptor stored in external memory is loaded as set values and a DMA transaction is executed using the loaded values. The DMAC contains a Next Link address and a Current Link address for each channel, and these addresses are used to set the descriptor address to be executed next and to indicate the descriptor address of the currently executed DMA transaction, respectively.

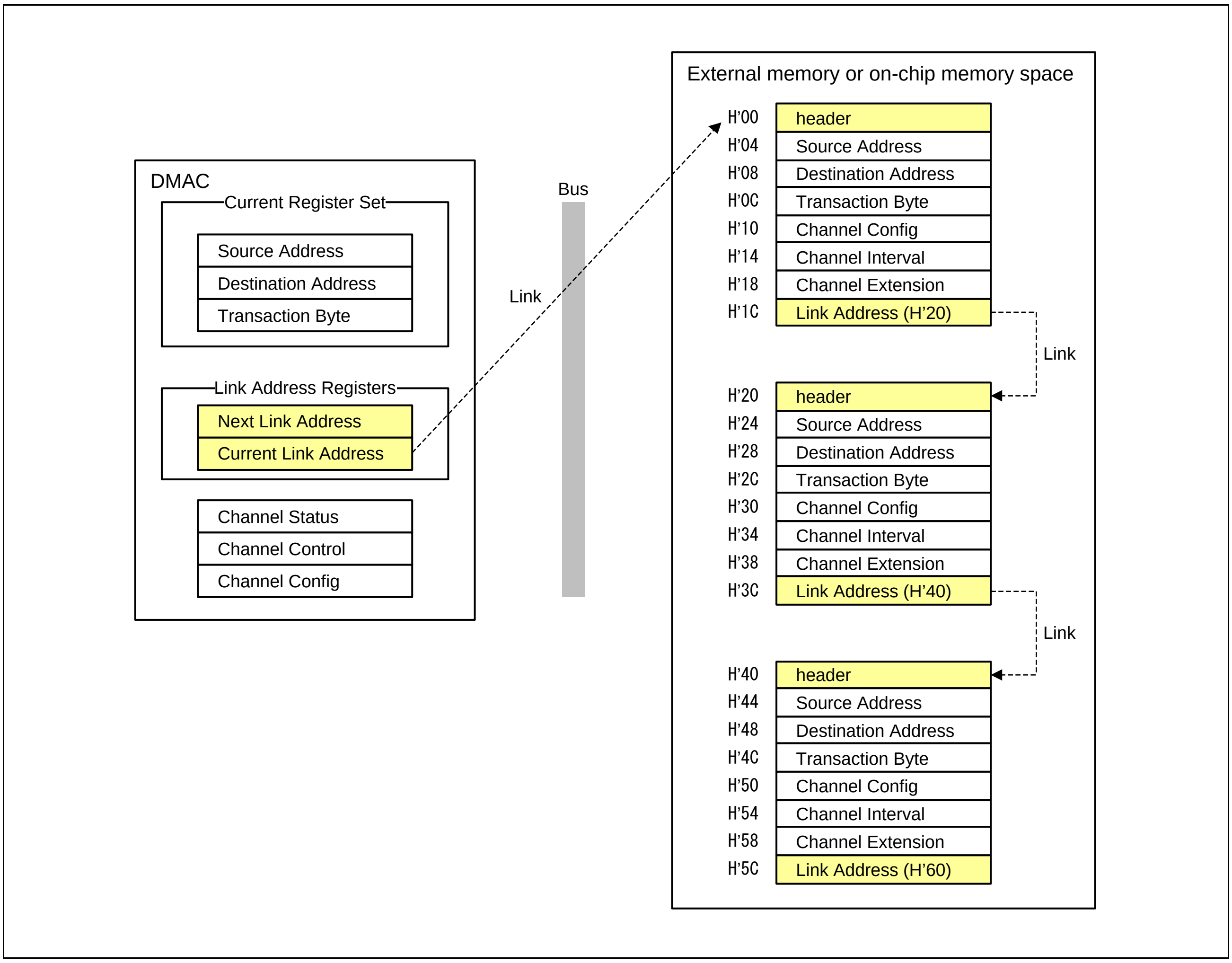


Figure 14.6 Link Mode Outline

(1) Operation Flow

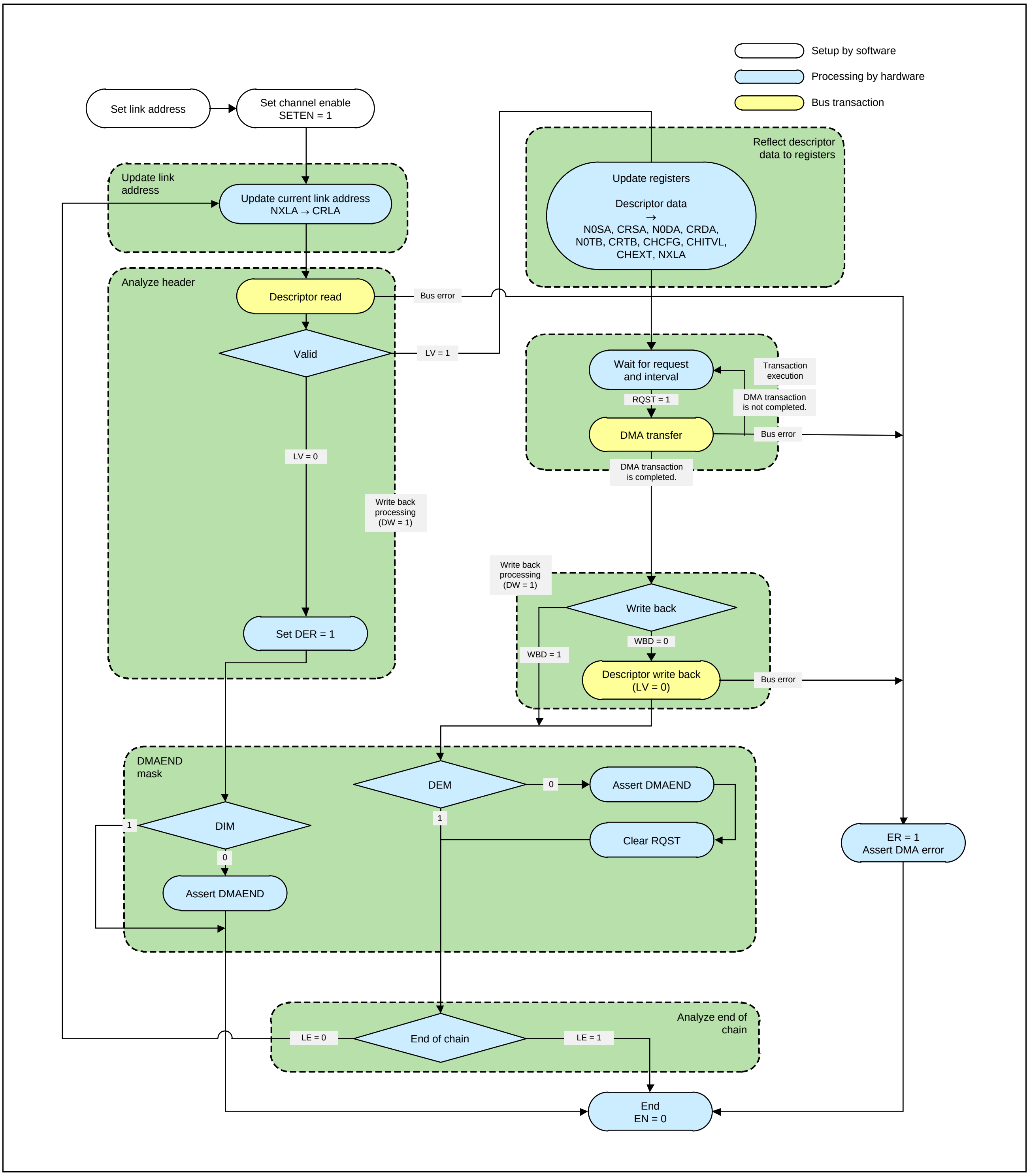


Figure 14.7 Link Mode Flow

<Explanation of the link mode flow>

1. Channel setting

The start address of the link destination is set in NXLA_n/nS.

2. Link address update

   When 1 is set in EN (1 is set in SETEN), the Link address set in NXLA_n/nS is loaded to CRLA_n/nS.

3. Descriptor load and header analysis

   The DMAC begins to load the descriptor and then analyzes the content of the header. When LV is 0, the DMAC discards the loaded descriptor and sets 1 in DER to end the operation (EN = 0). In this case, if 0 is set in DIM of the header, DMAEND is issued.

4. Descriptor setting

   The loaded descriptor is set in the Current register set and Channel register set. Also, the next link address is set in NXLA_n/nS.

5. DMA transaction

   A DMA transaction is executed according to the set values.

6. Header writeback

   When 0 is set in WBD of the header, the DMAC writes back the header with 0 set in its LV bit.

7. DMAINT mask

   When 0 is set in the DEM bit of CHCFG_n/nS, the DMA transfer end interrupt is issued.

8. Link end analysis

   When 1 is set in LE of the header, the operation is ended by clearing EN to 0, after transfer using the settings of the descriptor is completed. If the setting of LE is 0, the Current registers are then updated and loading of the next descriptor begins. The TEND signal is issued after the transfer of each descriptor.

(2) Register Setting

(a) Link mode setting

To use the link mode, set 1 in the DMS bit of the CHCFG_n/nS register.

Table 14.11 Link Mode Setting

DMS (CHCFG_n/nS)	Description
1	Operates in link mode. This bit cannot be changed using a descriptor.

(b) Link address setting

There are two registers that indicate a link address:

Next Link address register and Current Link address register.

To start the link mode, set a link address in the Next Link address register.

The Next Link address indicates the next link address after a descriptor is loaded. The Current Link address indicates the currently executed link address.

Table 14.12 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n/nS)	Sets and indicates the next link address. Before starting the link mode, set a link address in this register.
Current Link Address Register (CRLA_n/nS)	Indicates the currently executed link address. This register is read-only.

CAUTION

In link mode, the settings can be changed by reading a descriptor. It is not possible, however, to synchronize the change of the settings with a peripheral module request or external request. Therefore, when using a peripheral module request or external request, set AM, LVL, HIEN, LOEN, and SEL of the CHCFG_n/nS register before setting Enable and do not change any of these bits in the descriptor.

(3) Descriptor Setting

In a link address, prepare a descriptor with data arranged in the order shown below.

The DMAC reads the descriptor in burst mode.

(a) Descriptor data arrangement

Table 14.13 Descriptor Data Arrangement

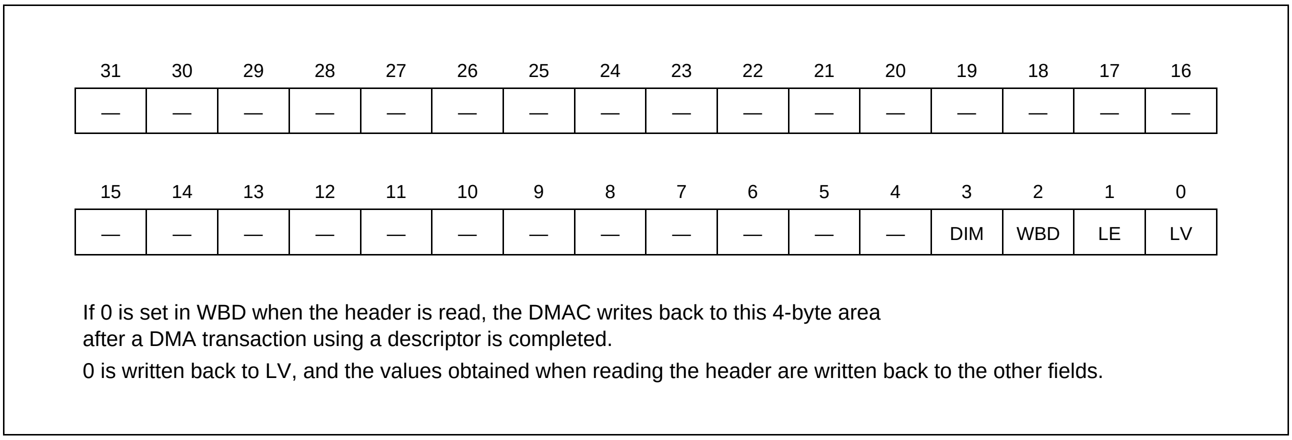
Address	Data	Remarks
Link address + H'00	header	
Link address + H'04	Source Address	
Link address + H'08	Destination Address	
Link address + H'0C	Transaction Byte	
Link address + H'10	Config	The register mode cannot be set.
Link address + H'14	Interval	
Link address + H'18	Extension	
Link address + H'1C	Next Link Address	

Note: As a link address, set an address aligned along the 32-bit boundary.

(b) header

The header indicates the status of the descriptor, as shown below.

The DMAC reads this area when a DMA transfer is started in link mode. Also, after a DMA transaction is completed, the DMAC writes back the transfer status to the area.



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	—	—	—	DIM	WBD	LE	LV

If 0 is set in WBD when the header is read, the DMAC writes back to this 4-byte area after a DMA transaction using a descriptor is completed.

0 is written back to LV, and the values obtained when reading the header are written back to the other fields.

Figure 14.8 Header Area

Table 14.14 Header Area

Bit Position	Bit Name	Meaning
31 to 4	—	—
3	DIM	Descriptor Interrupt Mask Sets whether to mask the DMA transfer end interrupt if 0 is set in LV when the header is loaded. 0: Issues a DMA transfer end interrupt. 1: Does not issue a DMA transfer end interrupt.
2	WBD	Write Back Disable Sets whether to mask LV bit writeback. When 1 is set in this bit, the DMAC does not perform writeback. 0: Writes the LV bit back to 0. 1: Does not write back the LV bit.
1	LE	Link End Indicates whether the link ends with the DMA transaction of this descriptor. Set 1 in this bit to indicate the end of the link. 0: The link continues. 1: The link ends.
0	LV	Link Valid Indicates whether this descriptor is valid. If 0 is set in WBD, the DMAC writes 0 in this bit after the DMA transaction written in the descriptor is executed. When setting the header, set 1 in this bit. 0: Descriptor invalid 1: Descriptor valid

(c) Descriptor data other than the header

The data items of the descriptor other than the header are the same as defined in the internal register specifications (note that the DMS bit of the CHCFG_n/nS register cannot be changed using the descriptor). For information about the internal register specifications, see **Section 14.4, Register Descriptions**.

For descriptor setting examples, see **Section 14.8, DMA Setting Examples**.

(d) CACHE settings for descriptor access

The CACHE settings for descriptor access can be set in LWCA and LDCA of the DMA control register (DCTRL_0_7/ 0_7S, DCTRL_8_15/8_15S). Make these settings as appropriate for the access destination in which the descriptor is prepared.

(e) Descriptor area and DMA transfer area

The following figure outlines the descriptor area and DMA transfer area that are accessed by the DMAC.

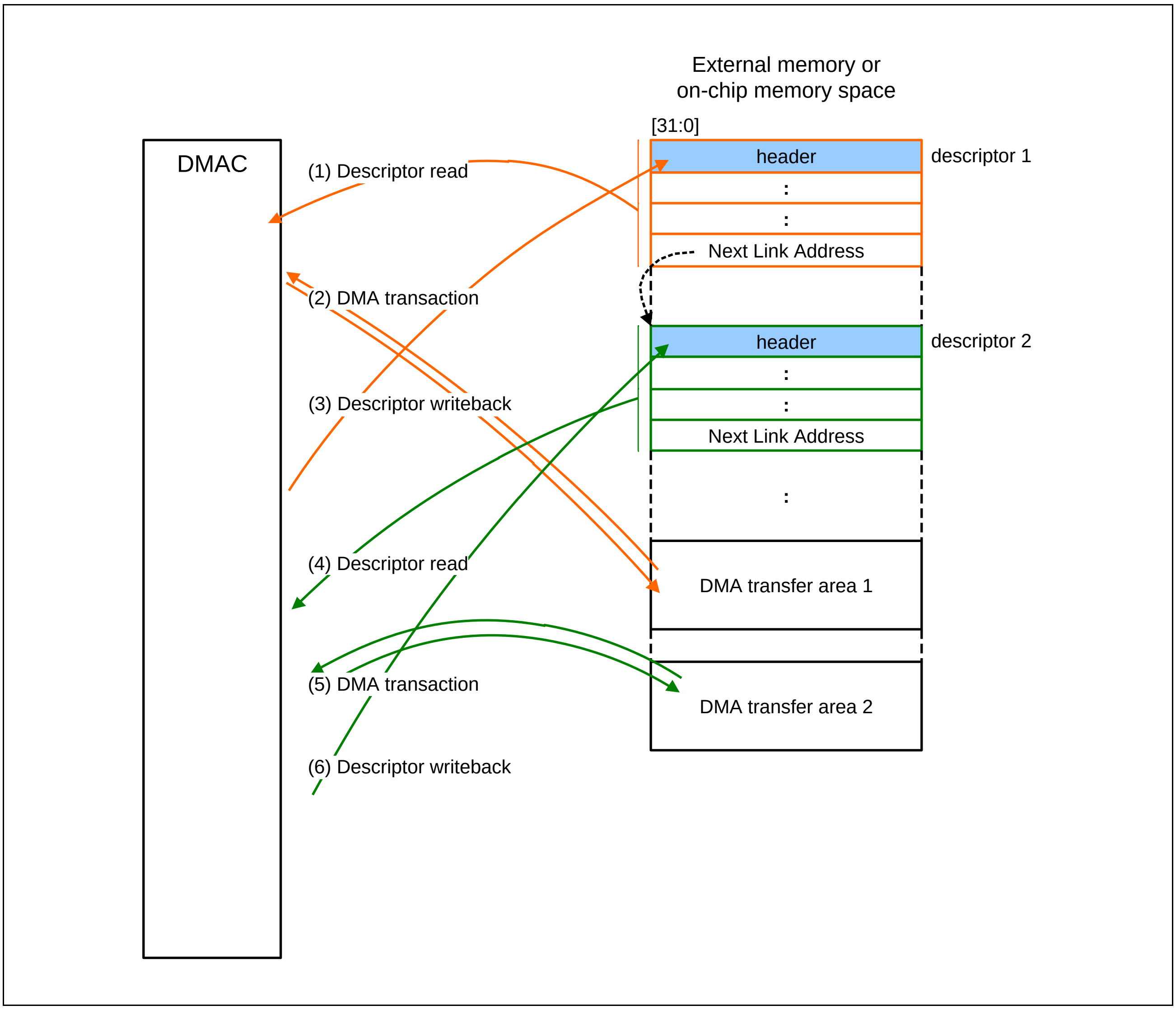


Figure 14.9 Outline of the Descriptor Area and DMA Transfer Area

1. Descriptor read

   The values set in the internal Next Link Address register are loaded to the Current Link Address register, and a descriptor is read from the external memory space (descriptor1) pointed to by the Current Link Address register.

2. DMA transfer

   When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.

3. Descriptor writeback

   When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor1, with 0 set in LV and the other bits containing the values read in <1>.

4. Descriptor read

   When 0 is set in the LE bit of the header in the last read descriptor (<1>), the next descriptor is read from the address (descriptor2) indicated by Next Link Address in the descriptor.

5. DMA transfer

   When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.

6. Descriptor writeback

   When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor2, with 0 set in LV and the other bits containing the values read in <4>.

4 through 6 are repeated.

When the header contains 1 in LE and 0 in WBD, the DMAC executes a DMA transfer using the settings of that descriptor, writes back data with 0 set in the LV bit of the header and ends the operation.

When the header contains 1 in both LE and WBD, the DMAC executes a DMA transfer using the settings of that descriptor and ends the operation (without writing back).

When the header contains 0 in LV, the DMAC ends the operation (without executing a DMA transfer).

(4) Descriptor Configuration Examples

In link mode, a descriptor can be configured as shown below.

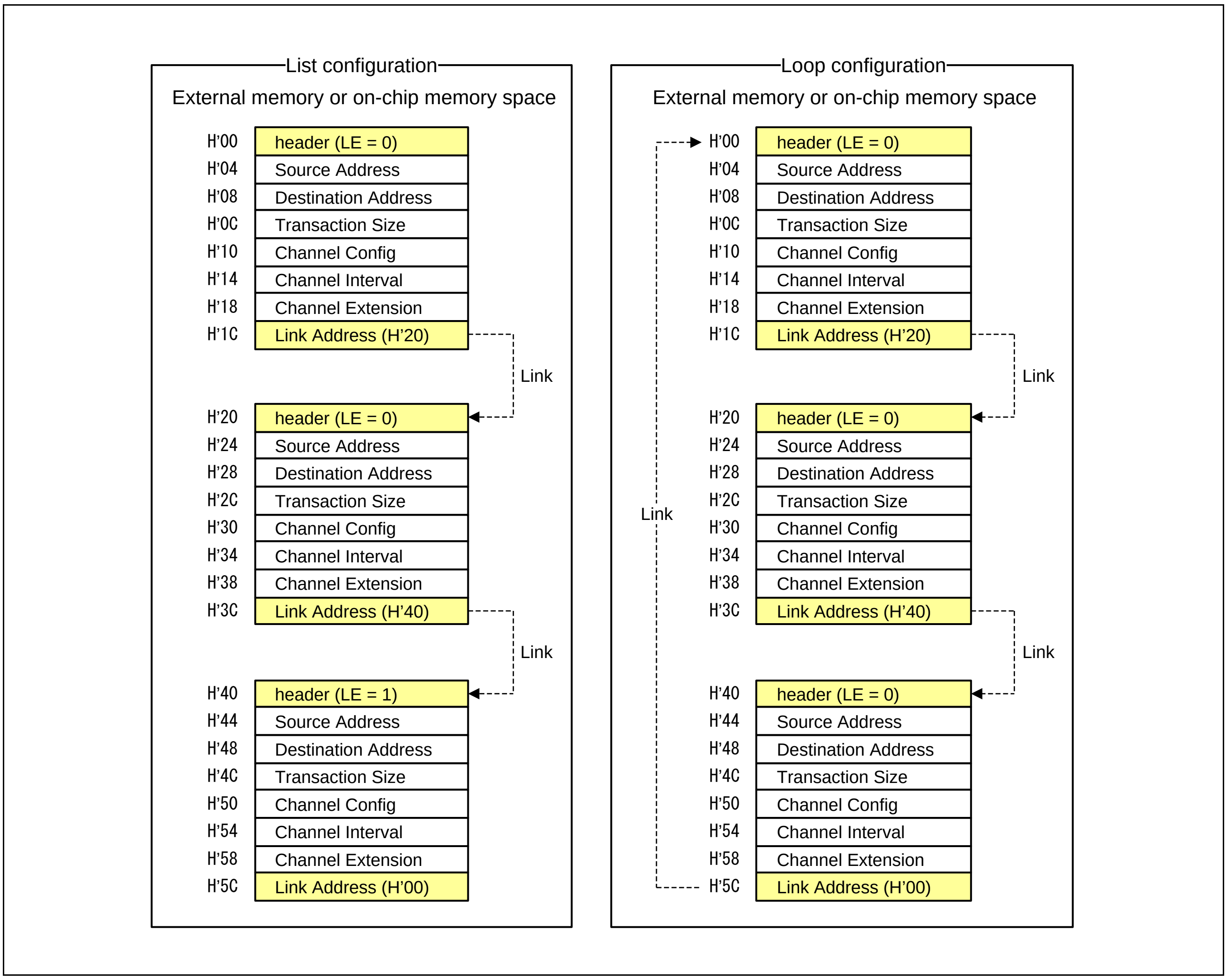


Figure 14.10 Descriptor Configuration Examples

- List configuration
  The link is ended by setting 1 in the LE bit of the header in the last descriptor.
- Loop configuration
  A descriptor can be created with a loop configuration, by setting the address of the top descriptor in the link address of the last descriptor. To end the loop, change the value of the LE bit of the header to 1 before the DMAC reads the descriptor, or follow the transfer suspension procedure.

14.7 DMA Transfer

The basic operation of DMA transfer is described here.

14.7.1 Transfer Mode

Two transfer modes are supported: single transfer mode and block transfer mode.

To select a transfer mode, set the TM bit of CHCFG_n/nS for each channel.

Table 14.15 Basic Transfer Setting

Transfer Mode	TM (CHCFG_n/nS)	Function
Single transfer	0	A single DMA transfer is executed in response to a DMAREQ.
Block transfer	1	In response to a DMAREQ, the DMAC continues to execute the transfer until the DMA transaction is completed.

(1) Single Transfer Mode

When a DMA transfer request is received, a DMA transfer is executed once in the direction indicated by REQD (source or destination). A DMA transfer is executed once each time a transfer request is received, and this operation continues until the number of bytes loaded to CRTB_n/nS is reached (arbitration between channels is accomplished for each DMA transfer).

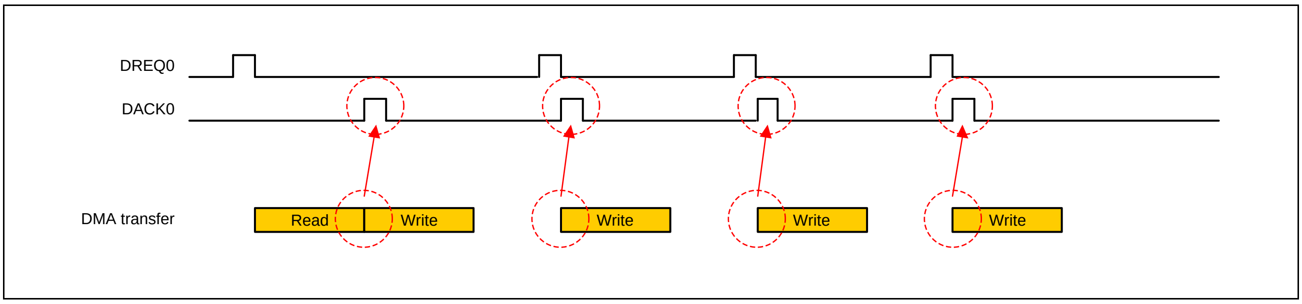


Figure 14.11 Single Transfer Mode (REQD = 1, SDS > DDS)

(2) Block Transfer Mode

Once a DMA transfer request is received, the DMAC continues to execute the transfer until data equivalent to the number of bytes loaded to the DMA transfer byte register (CRTB_n/nS register) is transferred (the DMA transaction is completed) (arbitration between channels is accomplished for each DMA transfer).

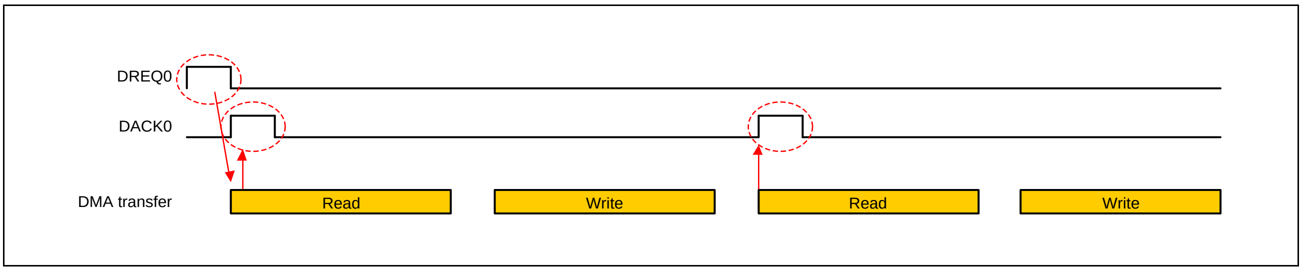


Figure 14.12 Block Transfer Mode (REQD = 0, SDS < DDS)

14.7.2 Priority Control for DMA Channels

Within channels 0 to 7 and 8 to 15, two priority control modes are supported: fixed priority mode and round robin mode. Only round robin mode is supported for priority control between the group of channels 0 to 7 and the group of channels 8 to 15. To select a priority control mode, use the PR bit of the DMA control register (DCTRL register). The fixed priority mode is selected when 0 is set in the PR bit, and the round robin mode is selected when 1 is set.

Read priority and write priority are controlled independently.

The DMAC issues transfer requests to different channels concurrently without waiting for the completion of any particular transfer and processes responses in the order it receives them. Therefore, the order in which the channels start transactions is not necessarily consistent with the order in which the transactions end.

Table 14.16 Priority Control Setting

Mode	PR (DCTRL)	Function	Purpose
Fixed priority	0	Requests are controlled based on the fixed order of priority for channels 0 to 7 and 8 to 15 (High: CH0 (CH8) > CH1 (CH9) > CH2 (CH10) > CH3 (CH11) > CH4 (CH12) > CH5 (CH13) > CH6 (CH14) > CH7 (CH15): Low).	Use this mode when the channels have a specific order of priority.
Round robin	1	Requests are controlled in a round robin fashion.	Use this mode to execute all requests evenly.

(1) Fixed Priority Mode

In fixed priority mode, the channels have a fixed order of priority in channels 0 to 7 and 8 to 15. Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

Immediately after a reset, the order of priority is as follows.

High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If there is a transfer request from DMA channel 0 in this state, a transfer is executed on DMA channel 0. After the transfer is completed, the order of priority is as follows.

High CH8 > CH0 > CH9 > CH1 > CH10 > CH2 > CH11 > CH3 > CH12 > CH4 > CH13 > CH5 > CH14 > CH6 > CH15 > CH7 Low

If a DMA transfer request occurs on multiple channels simultaneously, the DMA transfer request of the channel having the smallest channel number is given priority. The following figure shows an example where a DMA transfer request occurs on a channel having a higher priority while a DMA transfer is being executed in fixed priority mode.

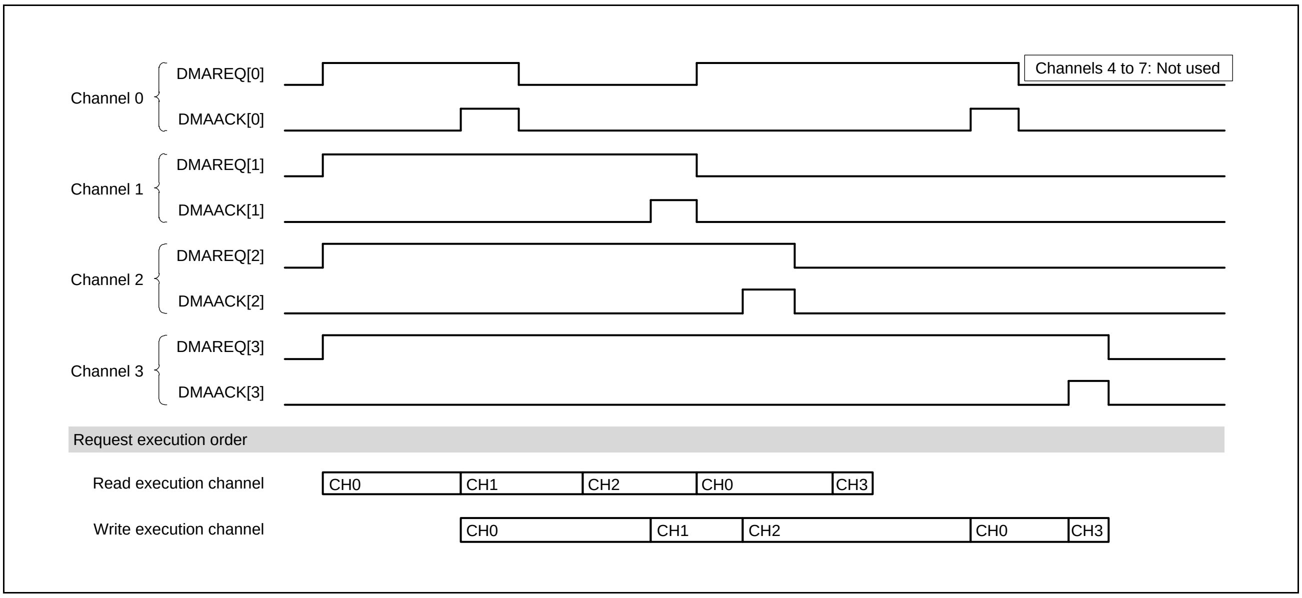


Figure 14.13 Fixed Priority Mode (Number of Channels = 4, REQD = 1)

(2) Round Robin Mode

In round robin mode, each time a transfer request is received from a channel in the group of channels 0 to 7 and the group of channels 8 to 15, the order of priority is changed in such a way that the channel that executed a transfer last has the lowest priority.
Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

Immediately after a reset, the order of priority is the same as that of the fixed priority mode, which is as follows.
High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If a transfer request is received from DMA channel 2 in this state, a transfer is executed on DMA channel 2. After the transfer is completed, the order of priority is as follows.
High CH8 > CH3 > CH9 > CH4 > CH10 > CH5 > CH11 > CH6 > CH12 > CH7 > CH13 > CH0 > CH14 > CH1 > CH15 > CH2 Low

The following figure shows an example where DMA transfers are executed in round robin mode.

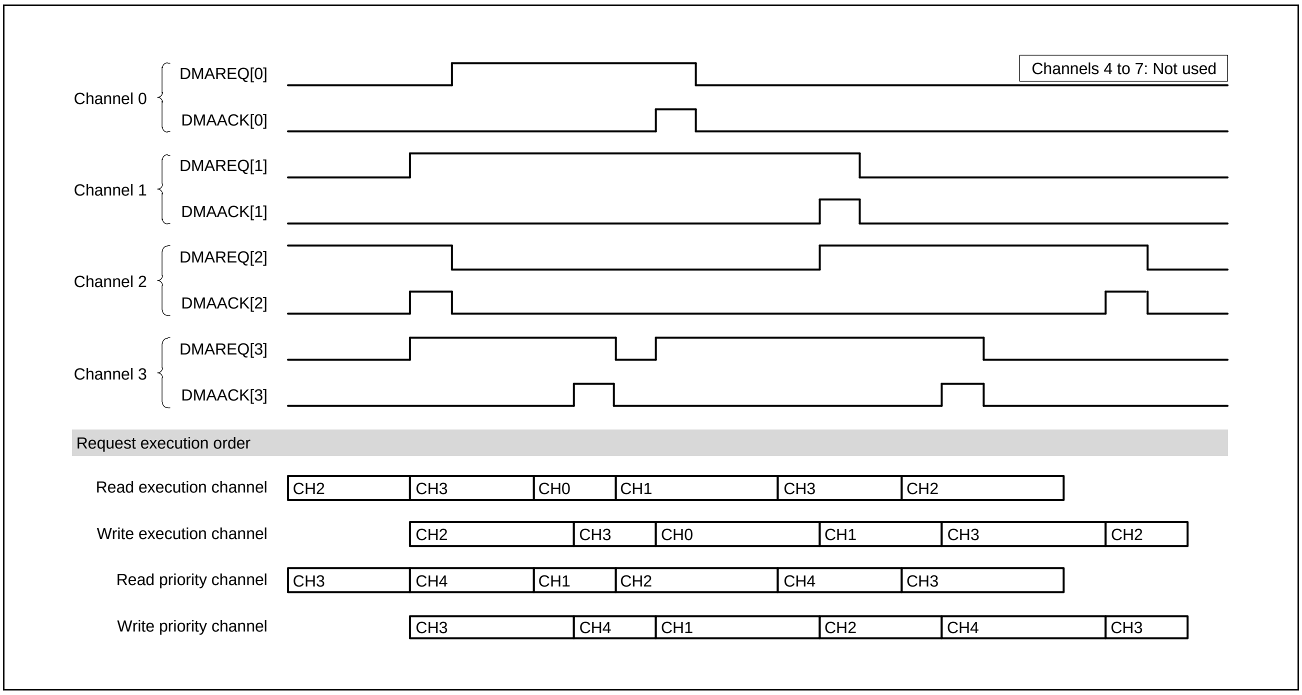


Figure 14.14 Round Robin Mode (Number of Channels = 4, REQD = 0)

The channel whose channel number is the number of the currently transferring channel + 1 gets to execute a DMA transfer next. If there is no transfer request from this channel, the channel whose channel number is the number of this channel + 1 gets to execute a DMA transfer.

14.7.3 DMA Transfer Request

Edge detection or level detection can be selected using the LVL bit of the CHCFG_n/nS register.

The HIEN and LOEN bits of the CHCFG_n/nS register are used to select either the rising edge or falling edge in the case of edge detection or either the high level or low level in the case of level detection.

When the transfer request is by an on-chip peripheral module, set the CHCFG_n/nS register according to **Table 14.3**.

(1) Edge Detection

Setting 0 in the LVL bit of the CHCFG_n/nS register enables edge detection.

(2) Level Detection

Setting 1 in the LVL bit of the CHCFG_n/nS register enables level detection.

14.7.4 DMA Transfer End Interrupt

The DMA transfer end interrupt is an interrupt request signal that indicates that a DMA transaction is completed. There is an independent DMA transfer end interrupt for each channel.

When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed, 1 is set in END of the CHSTAT_n/nS register. In this case, when 0 is set in DEM of the CHCFG_n/nS register, the DMA transfer end interrupt (DMAINTn_NS/S) is output (n = 0 to 15). (When writeback is performed in link mode, the signal is output after the writeback operation.)

When 0 is set in LV of the header in the read descriptor in link mode, 1 is set in DER of the CHSTAT_n/nS register. In this case, when 0 is set in DIM of the header, the DMA transfer end interrupt is output.

Table 14.17 Assertion Conditions of DMA Transfer End Interrupt

Source	Condition	DMA Transfer End Interrupt Mask Signal
DMA transaction end	When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed with an OKAY response (or after the writeback operation when writeback is performed in link mode)	DEM bit of the CHCFG_n/nS register
Descriptor invalid	When 0 is set in LV of the header in the read descriptor in link mode while 0 is set in DIM of the header	DIM bit of the header

14.7.5 DMA Error Interrupt

If an error response is received for a DMA transfer or descriptor access, the DMAC regards it as an error and stops the transfer. Upon receiving an error response, the EN bit of the CHSTAT_n/nS register of transferring channel n is cleared to 0 and 1 is set in the ER bit (n = 0 to 15). Also, the DMA error interrupt (DMAERR_NS/S) is output.

The DMA error interrupt cannot be masked.

Once an error occurs, the data of the whole transfer cannot be guaranteed. Be sure to start the transaction again from the beginning by following the procedure below.

1. Set 1 in the SWRST bit of the CHCTRL_n/nS register.
2. Set each register again.

14.7.6 Interval Count Function

The interval at which a DMA transfer is executed can be adjusted by setting the ITVL bit of the channel interval register (CHITVL_n/nS). This function is intended to prevent the DMA controller from occupying the bus all the time.

When a read or write operation is completed, a countdown starts from the value set in CHITVL_n/nS. The next internal request is not executed until the count value reaches 0.

The following figure shows an example of how this works.

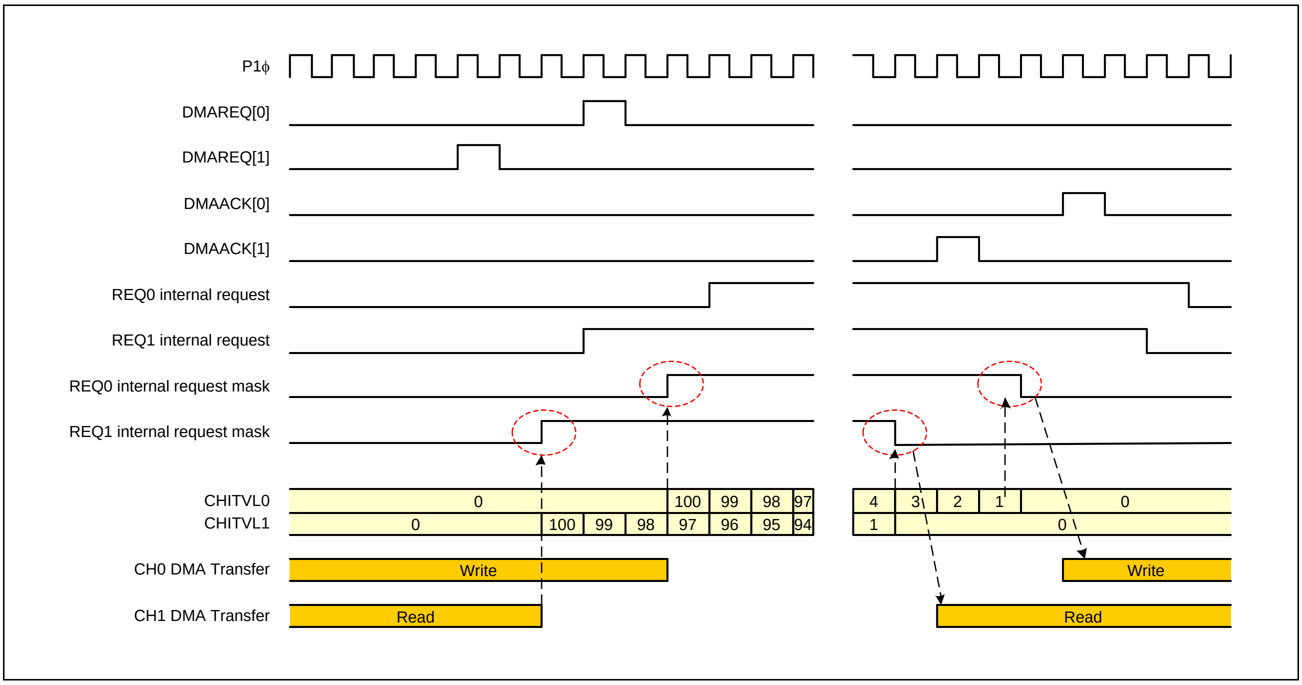


Figure 14.15 Interval Count

14.7.7 Difference in Operation Due to the Transfer Size

(1) When the Source Transfer Size Is Smaller

When the read of data equivalent to the destination data size is completed, the data is written to the destination.

The following figure shows a timing chart where the source transfer size is 8 bits and the destination transfer size is 32 bits (in the case of rising edge detection).

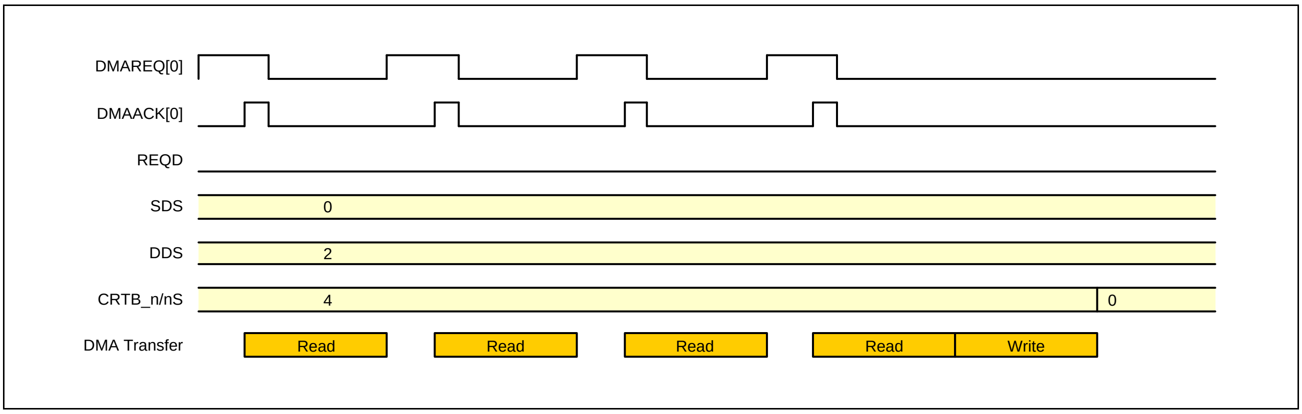


Figure 14.16 When the Source Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 0, SDS < DDS in CHCFG_n/nS)

(2) When the Destination Transfer Size Is Smaller

Since the source transfer size is larger, multiple destination writes occur after a single source read. The following figure shows a timing chart where the source transfer size is 64 bits and the destination transfer size is 16 bits (in the case of rising edge detection) (1 is set in REQD of the CHCFG_n/nS register).

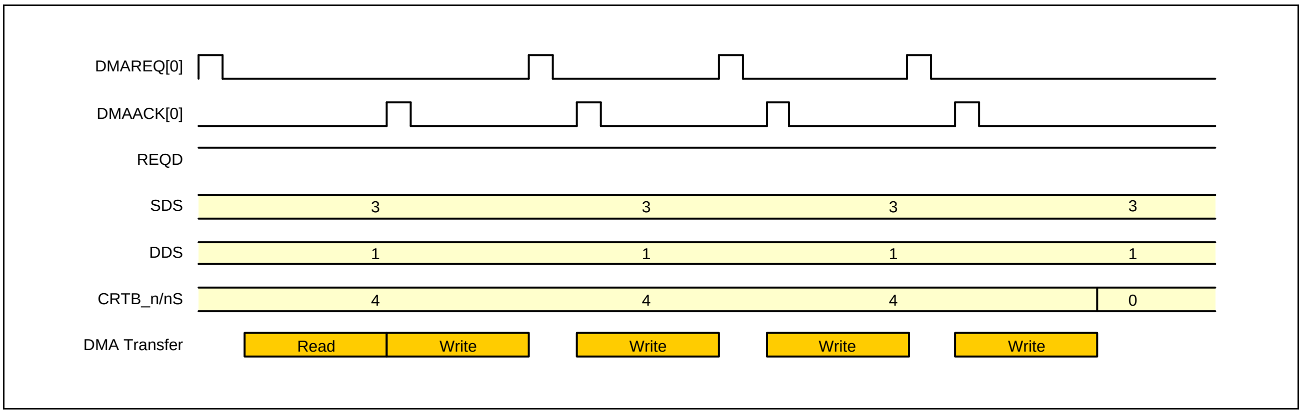


Figure 14.17 When the Destination Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 1, SDS > DDS in CHCFG_n/nS)

(3) When the Source Transfer Size Is the Same as the Destination Transfer Size

Every time a DMA transfer request is detected, a source read and a destination write occur.

The following figure shows a timing chart where the source transfer size and the destination transfer size are both 8 bits (in the case of rising edge detection, with 1 set in REQD of the CHCFG_n/nS register).

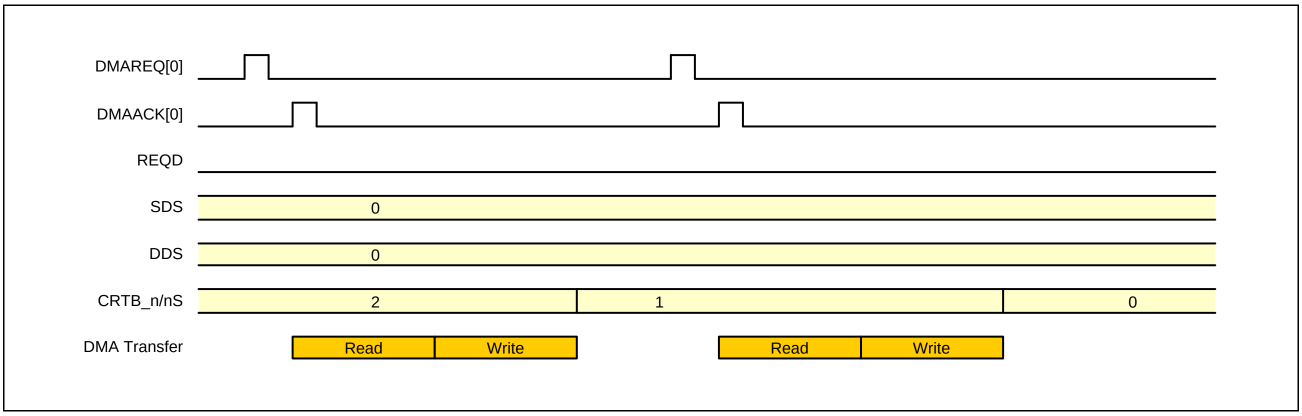


Figure 14.18 When the Source Transfer Size Is the Same as the Destination Transfer Size (LVL = 0, HIEN = 1, REQD = 0, SDS = DDS in CHCFG_n/nS)

14.7.8 Transfer Status

The channel status register indicates the status of DMA transfer execution on a channel.

(1) Suspend

A DMA transfer can be suspended by using the SETSUS bit of CHCTRL_n/nS. In this case, if an ongoing bus cycle exists, the DMAC waits for that cycle to end before suspending the transfer. Writing 1 in the CLRSUS bit restores the DMA transfer from the suspend status.

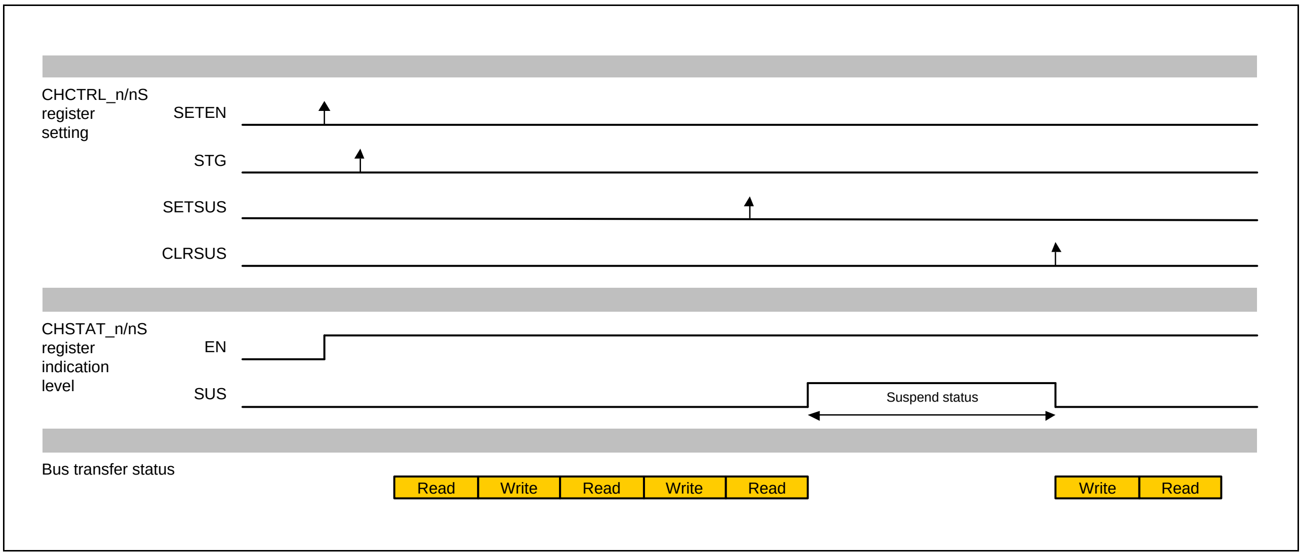


Figure 14.19 DMAC Suspend Status (Auto Request/Block Transfer)

In the above case, the DMA transfer is suspended after the read transfer is completed.

If there is any ongoing DMA transfer, the suspend status starts when that transfer is completed. To make sure that the transfer is suspended, read the CHSTAT or DSTAT_SUS register, after setting SETSUS, and check that 1 is set in the SUS bit for the relevant channel.

(2) Transfer Stop

If 1 is written to CLREN while a DMA transaction is in progress, the DMA transaction for the corresponding channel can be stopped. For the post-stop processing, two modes are supported: one sweeps out the data remaining in the buffer when the transaction is stopped (SBE = 1) and the other does not (SBE = 0). One of these modes can be selected using the SBE bit of the CHCFG_n/nS register. By default, SBE is set to 0.

When this sweep mode is enabled and CLREN is set to 1, and if a DMA transaction is stopped with data remaining in the DMAC buffer, the transaction is completed after the DMAC sweeps the data.

(a) Transfer Stop (Buffer Sweep Disabled - SBE = 0)

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. The stop timing depends on the value set in REQD. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMA internal status before setting the next transfer.

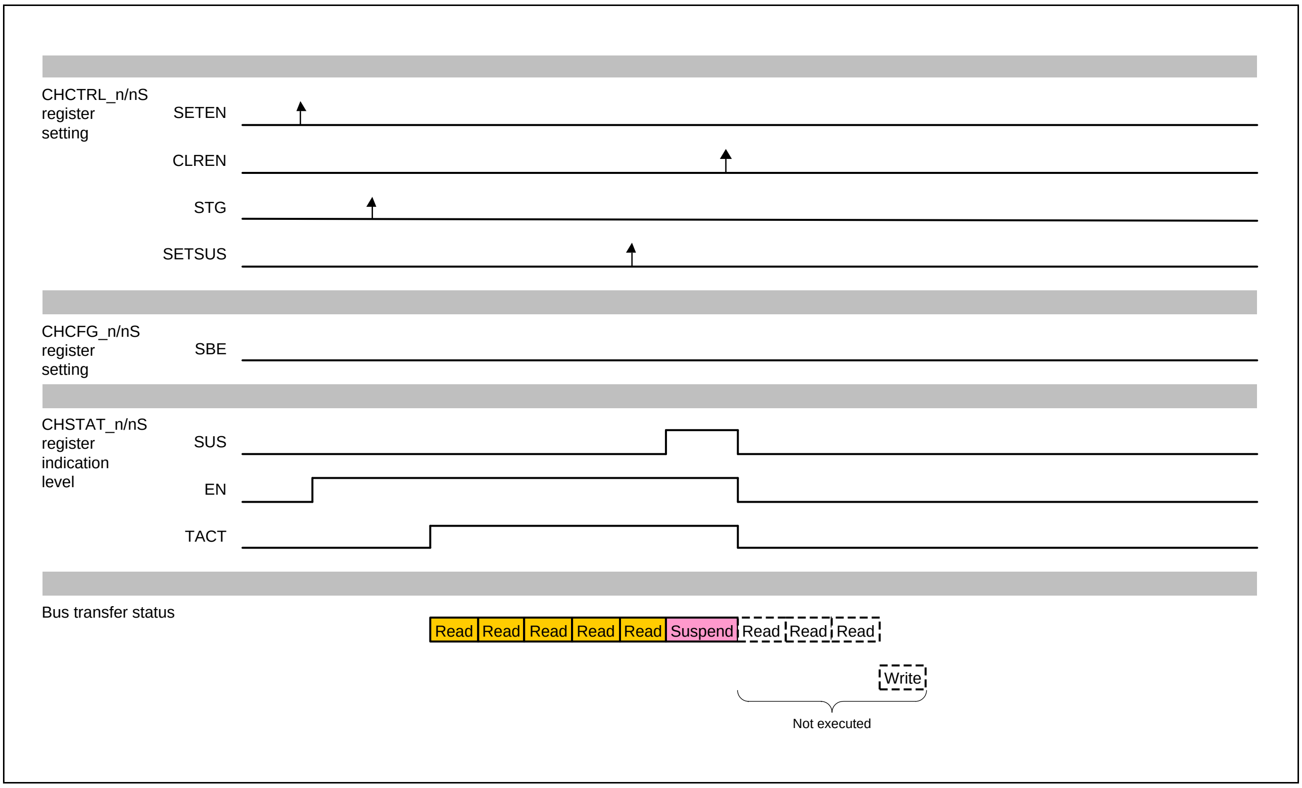


Figure 14.20 DMAC Transfer Stop

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If an ongoing DMA transfer is stopped before it is completed, the DMA transfer end interrupt is not asserted.
- If 0 is set in REQD, the DMA transfer is stopped when the next read is completed. (If the buffer contains any data that can be written, the DMA transfer is stopped after the data is written.)
- If 1 is set in REQD, the DMA transfer is stopped when the next write is completed.

(b) Transfer Stop (Buffer Sweep Enabled - SBE = 1)

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. When 0 is set in REQD, the DMA transfer is stopped after the DMAC sweeps (writes) the already read data. If 1 is set in REQD to use hardware requests, do not use the sweep mode. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMAC internal status before setting the next transfer.

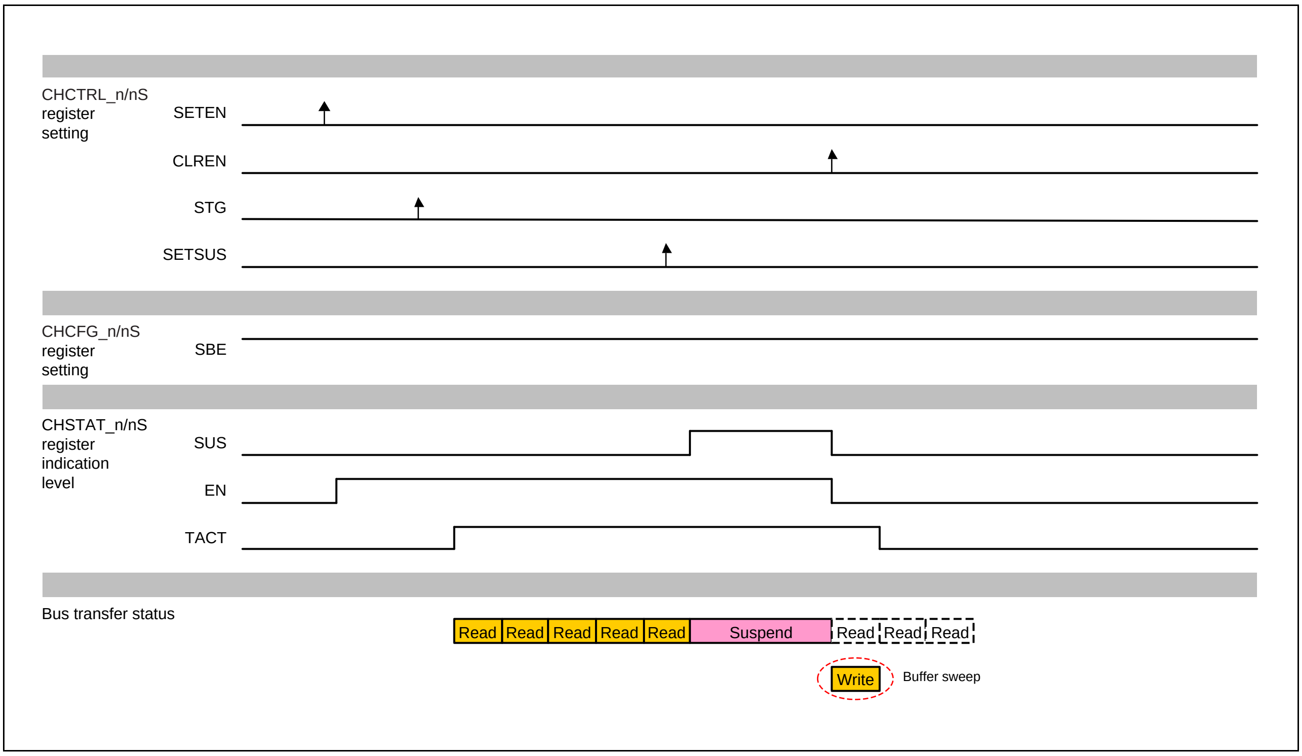


Figure 14.21 DMA Transfer Stop (Buffer Sweep Mode)

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If a transfer is stopped in sweep mode (SBE = 1) during the fifth read transfer by setting SETSUS and then CLREN, the read data is written before the DMA transfer is stopped.

(c) Channel Stop Check Method

Even when the EN bit is cleared to 0, the DMA transfer cannot be stopped immediately, if the bus is already executing the transfer. Therefore, in order to make sure that the DMAC has been brought to a complete stop, check that the EN bit and TACT bit are both set to 0.

(d) Transfer Stop Procedure

The transfer stop procedure is described below.

1. Set 1 in SETSUS of CHCTRL_n/nS.
2. Repeat polling until the SUS bit of CHSTAT_n/nS is set to 1. (If EN is already set to 0, the DMAC has already been stopped. Go to step 6.)
3. Set 1 in CLREN of CHCTRL_n/nS.

4. When 0 is set in SBE, the transfer is stopped according to the value of REQD. When 1 is set in SBE, the sweep mode is enabled. When 1 is set in SBE, set 0 in REQD.
5. Read CHSTAT_n/nS to check that 0 is set in the TACT bit. When TACT is set to 0, it means that the DMAC has been brought to a complete stop. When TACT is set to 1, repeat polling until this bit is set to 0.
6. To execute the next DMA transfer after stopping a transfer, be sure to set 1 in the SWRST (software reset) bit of CHCTRL_n/nS before the next DMA transfer starts.

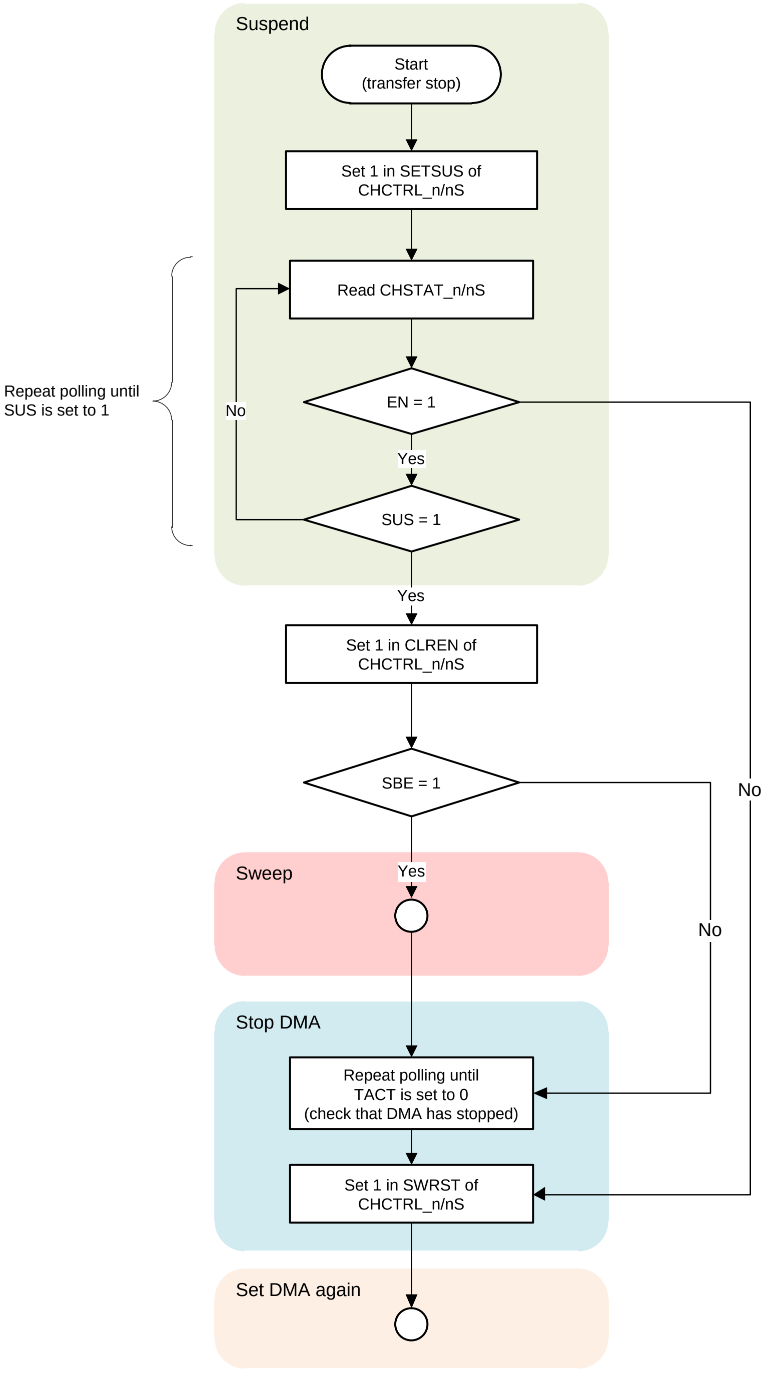


Figure 14.22 Transfer Stop Flow

14.8 DMA Setting Examples

Setting examples applicable when DMA transfer is executed using the direct memory access controller are shown in the following.

The transfer conditions for these setting examples are as follows.

Table 14.18 Transfer Condition List for DMA Transfer Setting Examples

	DMA Mode	Transfer Mode	Transfer Request
Setting example 1	Register	Single	Hardware
Setting example 2	Register	Block	Software
Setting example 3	Register (continuous execution)	Block	Software
Setting example 4	Link	Block	Software

For details of the settings, see the individual setting examples.

14.8.1 Setting Example 1 (Register Mode/Hardware Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.19 DMA Transfer Setting Example 1

Item		Description	
Channel used		3	
DMA mode		Register	
Transfer mode		Single transfer	
Register set used		Next0	
Source/destination		Source	Destination
	Start address	H’11110000	H’22220000
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
DMA transfer byte count		64 bytes	
DMA transfer request		Rising edge detection by hardware	
DMAACK signal		Level output during read	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	

Setting example 1

N0SA = H’11110000 (source address)
N0DA = H’22220000 (destination address)
N0TB = H’00000040 (transfer byte count)

CHCFG = H’00022123 (configuration)
CHITVL = H’00000000 (interval)
CHEXT = H’00000000 (CACHE setting)

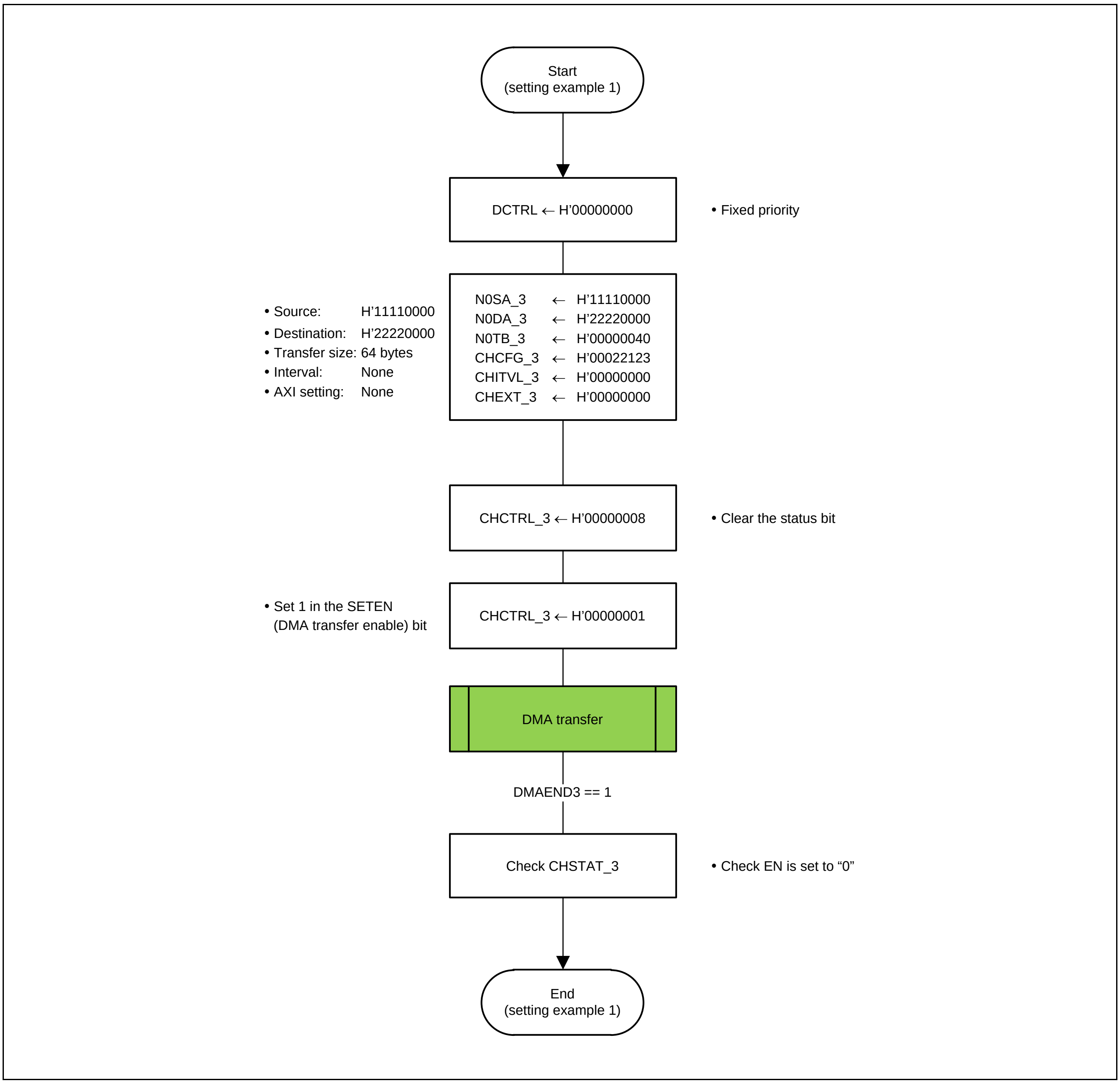


Figure 14.23 Setting Example 1

14.8.2 Setting Example 2 (Register Mode/Software Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.20 DMA Transfer Setting Example 2

Item		Description	
Channel used		2	
Priority control		Round robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set used		Next1	
Source/destination		Source	Destination
	Start address	H’0FFFE000	H’33330000
	Address direction	Increment	Increment
	Data size	8 bits	256 bits
DMA transfer byte count		128 bytes	
DMA transfer request		Auto request	
DMAACK signal		Masked	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	

Setting example 2

DCTRL = H’00000001 (DMA setting)

N1SA = H’0FFFE000 (source address)
N1DA = H’33330000 (destination address)
N1TB = H’00000080 (transfer byte count)

CHCFG = H’10450402 (configuration)
CHITVL = H’00000000 (interval)
CHEXT = H’00000000 (CACHE setting)

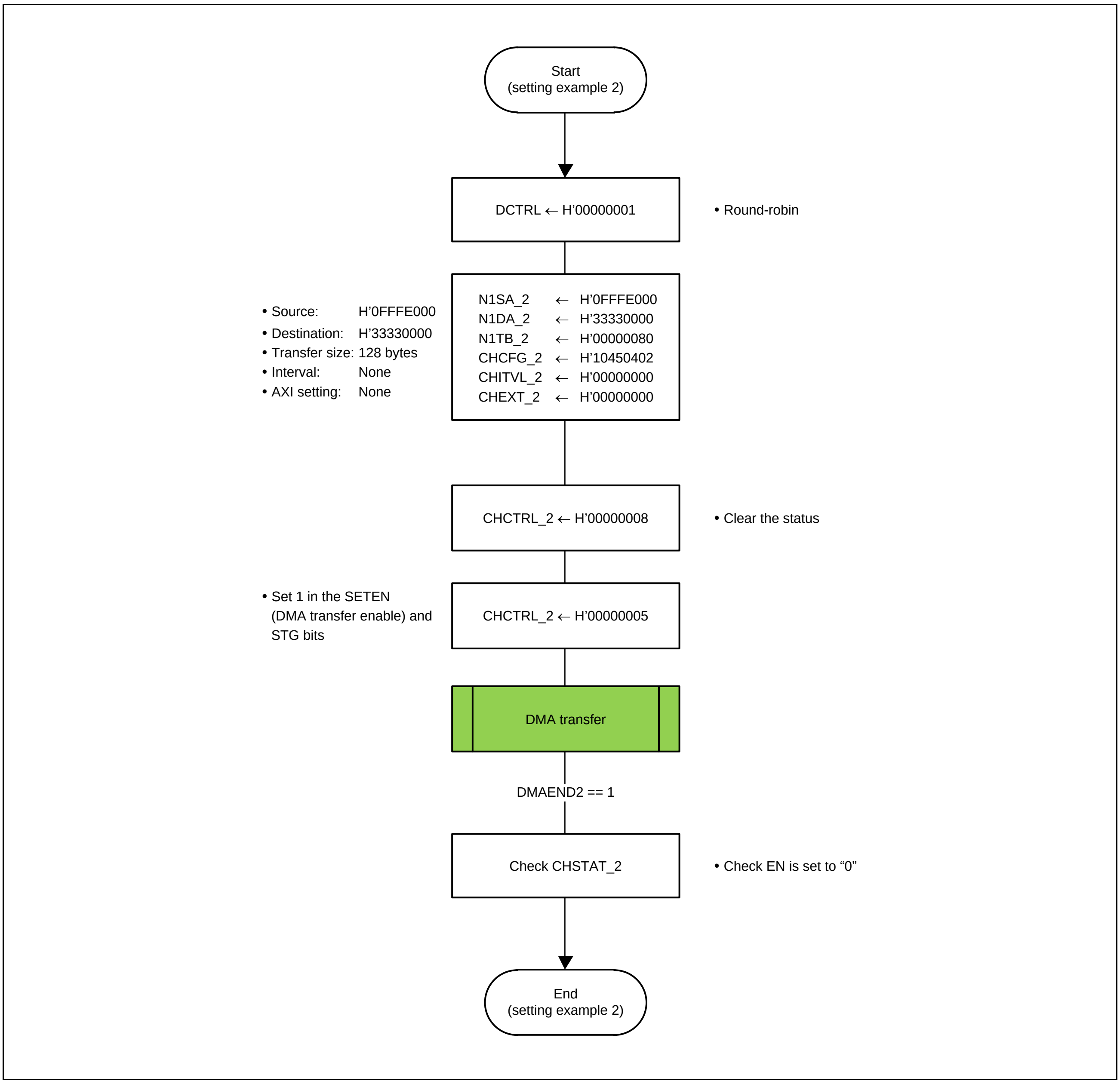


Figure 14.24 Setting Example 2

14.8.3 Setting Example 3 (Register Mode/Continuous Execution)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.21 DMA Transfer Setting Example 3

Item		Description	
Channel used		1	
Priority control		Round robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set used		Use Next0 and then Next1 continuously	
Next0		Source	Destination
	Start address	H’11110000	H’33330000
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	DMA transfer byte count	512 bytes	
Next1		Source	Destination
	Start address	H’22220000	H’44440000
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	DMA transfer byte count	2048 bytes	
DMA transfer request		Auto request	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Mask the DMA transfer end interrupt upon completion of Next0	
CACHE setting		Default value	

Setting example 3

DCTRL = H’00000001 (DMA setting)

N0SA = H’11110000 (source address)
N0DA = H’33330000 (destination address)
N0TB = H’00000200 (transfer byte count)

N1SA = H’22220000 (source address)
N1DA = H’44440000 (destination address)
N1TB = H’00000800 (transfer byte count)

CHCFG = H’61762001 (configuration)
CHITVL = H’00000000 (interval)
CHEXT = H’00000000 (CACHE setting)

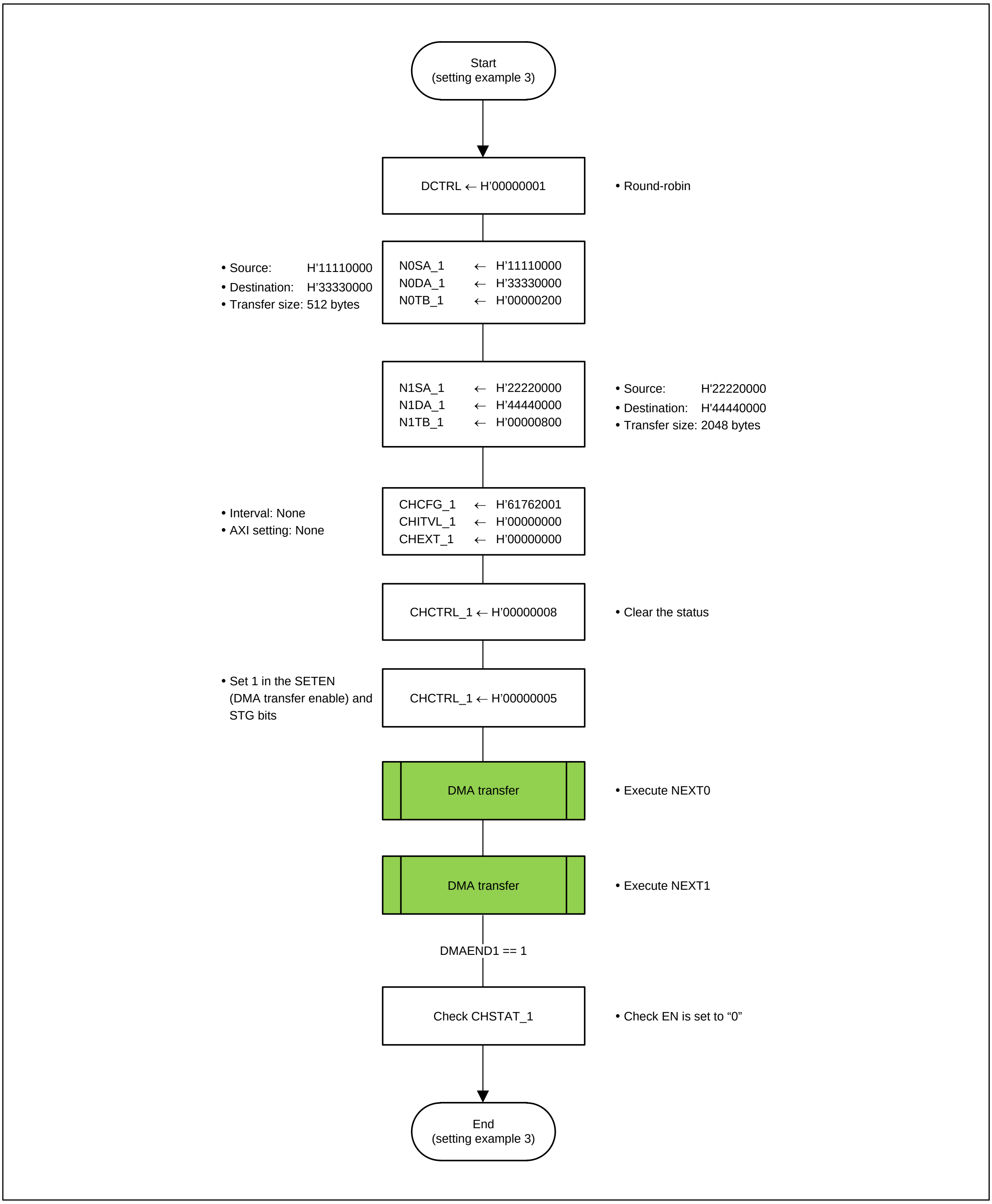


Figure 14.25 Setting Example 3

14.8.4 Setting Example 4 (Link Mode)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 14.22 DMA Transfer Setting Example 4

Item	Description
Channel used	0
Priority control	Round robin
DMA mode	Link
Transfer mode	Block transfer
Register set used	—
Descriptor start address	H'00001000

Table 14.23 DMA Transfer Setting Example 4 (Descriptor 1)

Item		Description	
Descriptor start address		H’00001000	
Next descriptor start address		H’00002000	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H’11110000	H’33330000
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
	DMA transfer byte count	2048 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Available (LE = 0)	
	Descriptor valid	Valid (LV = 1)	

Table 14.24 DMA Transfer Setting Example 4 (Descriptor 2)

Item		Description	
Descriptor start address		H’00002000	
Next descriptor start address		H’00005000	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H’44440000	H’55550000
	Address direction	Increment	Increment
	Data size	64 bits	256 bits
	DMA transfer byte count	1024 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Available (LE = 0)	
	Descriptor valid	Valid (LV = 1)	

Table 14.25 DMA Transfer Setting Example 4 (Descriptor 3)

Item		Description	
Descriptor start address		H’00005000	
Next descriptor start address		—	
Transfer mode		Block transfer	
Next0		Source	Destination
	Start address	H’77770000	H’AAAA0000
	Address direction	Increment	Increment
	Data size	512 bits	512 bits
	DMA transfer byte count	4096 bytes	
DMA transfer request		Auto request trigger (STG)	
DMAACK signal		Not output	
DMA transfer end interrupt mask		Not masked	
CACHE setting		Default value	
header			
	DMA interrupt when LV = 1	Issued (DIM = 0)	
	LV writeback	Done (WBD = 0)	
	Next link address	Not available (LE = 1)	
	Descriptor valid	Valid (LV = 1)	

Setting example 4

DCTRL= H'00000001 (DMA setting)
NXLA = H'00001000 (descriptor start address)
CHCFG = H'80000000 (configuration)

Table 14.26 Descriptor Setting

	Descriptor 1	Descriptor 2	Descriptor 3
header	H’00000001	H’00000001	H’00000003
SA (Source Address)	H’11110000	H’44440000	H’77770000
DA (Destination Address)	H’33330000	H’55550000	H’AAAA0000
TB (Transaction Byte)	H’00000800	H’00000400	H’00001000
CFG (Configuration)	H’81422008	H’81453008	H’80466008
ITVL (Interval)	H’00000000	H’00000000	H’00000000
EXT (Extension)	H’00000000	H’00000000	H’00000000
NXLA (Next Link Address)	H’00002000	H’00005000	H’00000000

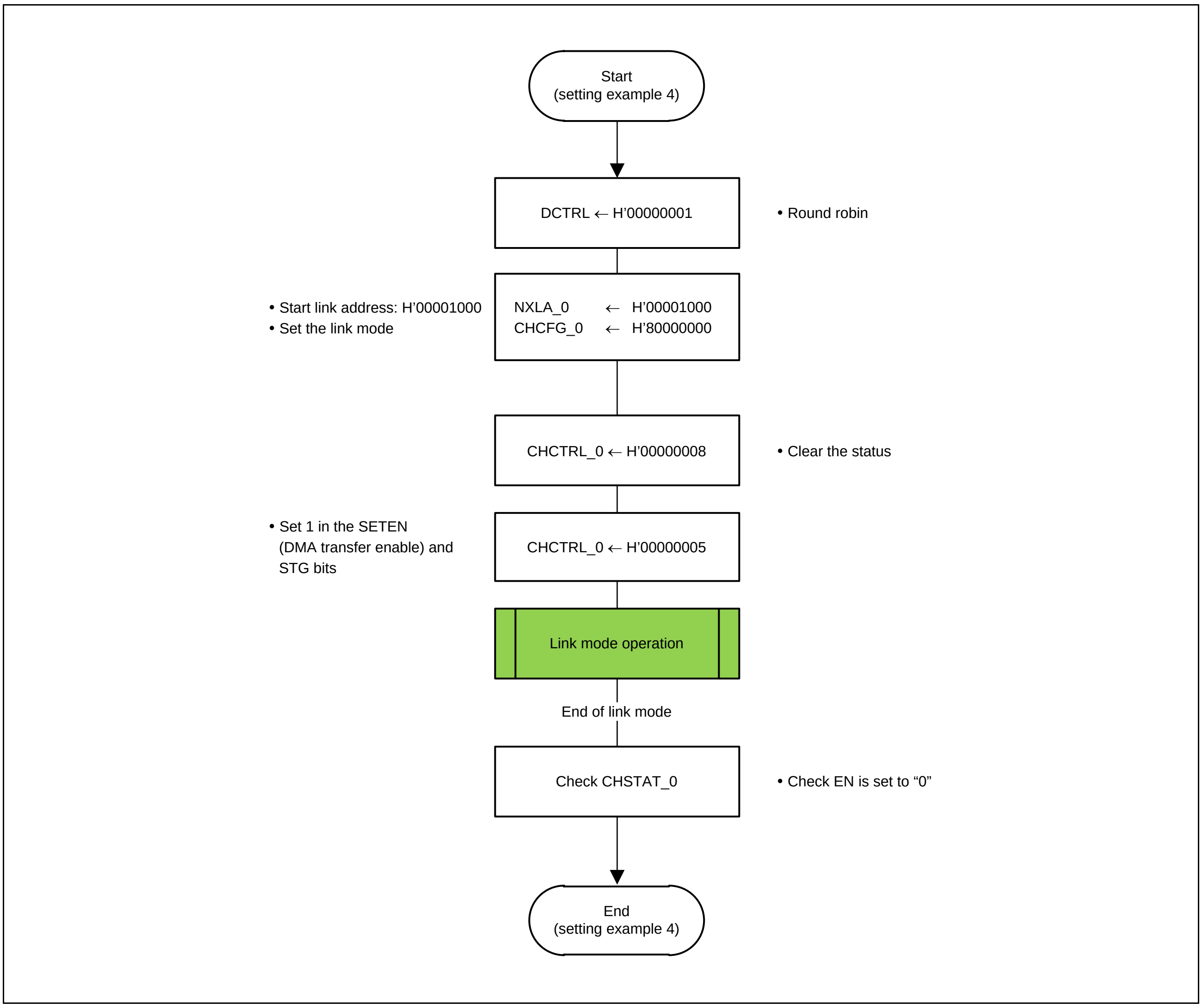


Figure 14.26 Setting Example 4

14.8.5 Next Register Set Continuous Execution Setting

The following figure shows the flowchart for executing DMA transfers continuously by using two Next register sets in register mode. While a DMA transaction is being executed using one Next register set, the other Next register set is set in order to continue to execute DMA transfers.

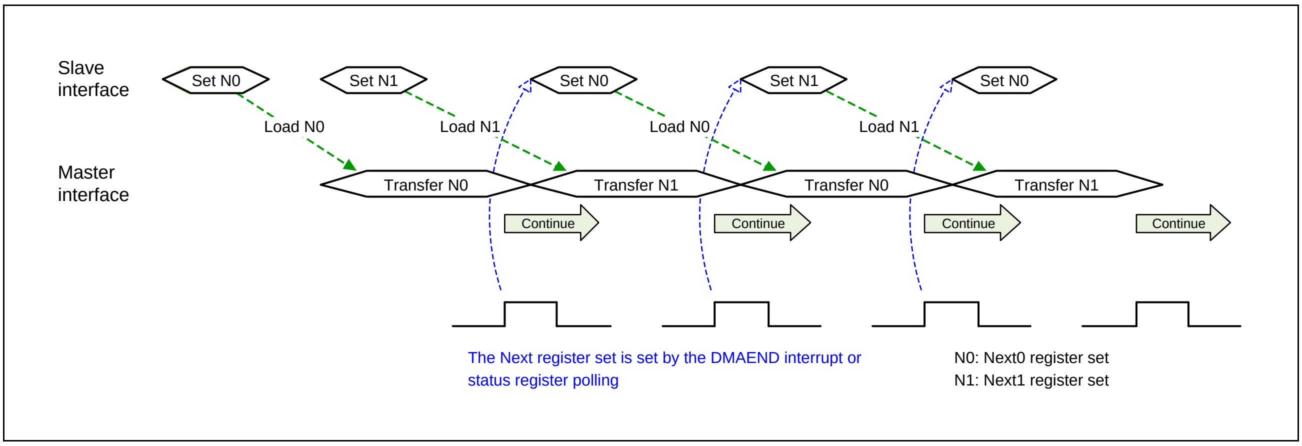


Figure 14.27 Image of Next Register Set Continuous Execution

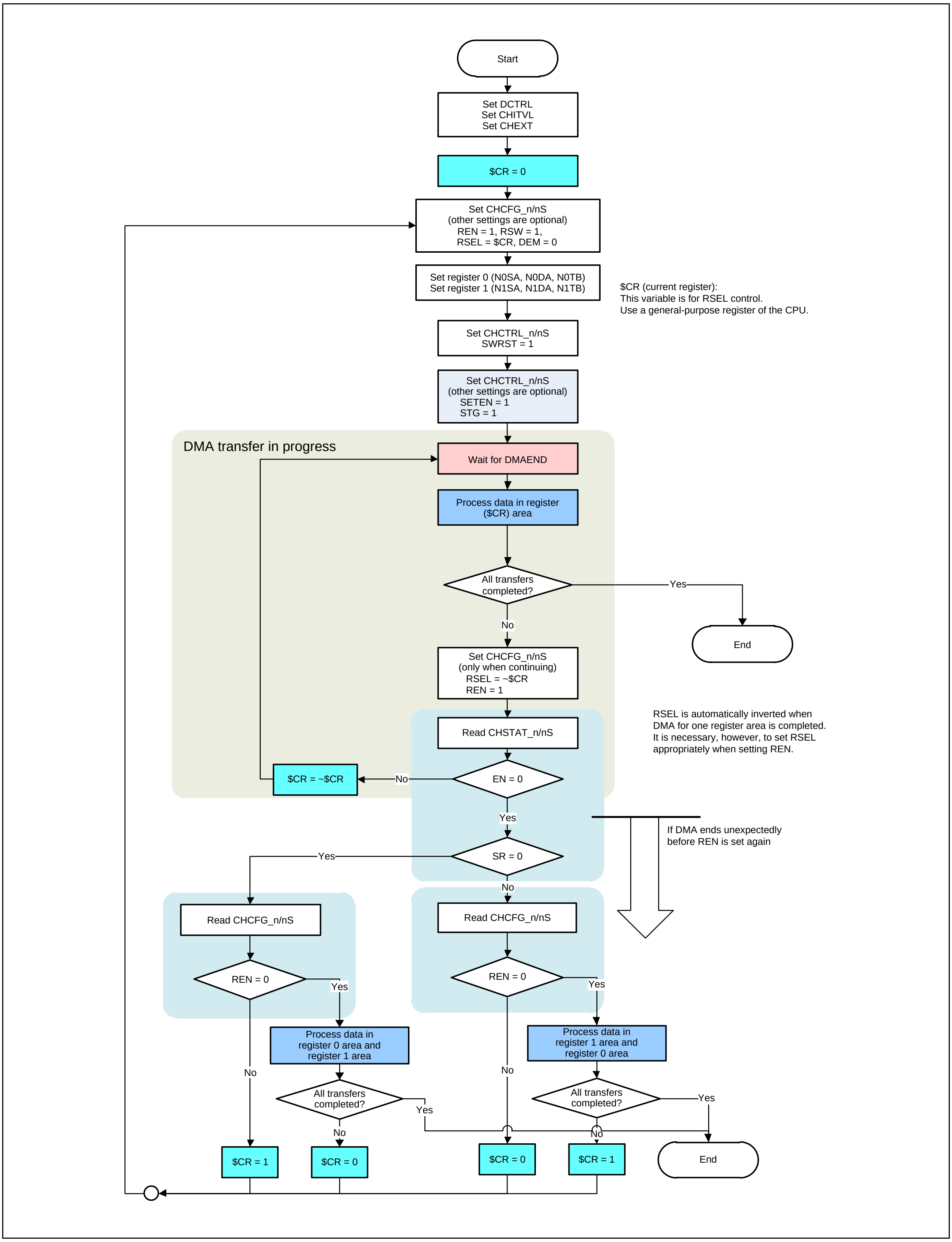


Figure 14.28 Example of Continuous DMA Execution by Using a Next Register Set

- Supplementary information

  First, save the data of the register sets to be used for DMA transfers (0 (N0SA, N0DA, and N0TB) and 1 (N1SA, N1DA, and N1TB)) to a general-purpose register of the CPU (the values of this register is referred to as $CR for the sake of convenience).

  Each time the DMA transfers for one register set are completed (the DMA transfer end interrupt is output), REN is automatically cleared to 0. In order to continue to execute DMA transfers, it is necessary to set REN of the CHCFG_n/nS register every time the DMA transfer end interrupt is asserted. This register also contains the RSEL bit, and the value of this bit needs to be set appropriately as well. Therefore, use $CR.

  In this mode, two Next register sets are executed continuously. However, if CLREN is not set before the DMA transaction is completed (the next DMA transfer end interrupt is output), continuous execution stops. In this case, how much of data has been transferred can be checked by reading the SR and EN bits of the CHSTAT_n/nS register and the REN bit of the CHCFG_n/nS register. To restart the DMA transaction, follow the flowchart shown above.

15. System Counter (SYC)

SYC generates the count value used by 3DGE and the generic timer built into Cortex-A55.

It uses the Timestamp generator, which is one of the components of Arm CoreSight SoC-400, to generate a 64-bit count value. For more information on the Timestamp generator, please refer to *the ARM CoreSight SoC-400 Technical Reference Manual*.

15.1 Overview

15.1.1 Features

- Count by 24 MHz clock (SYC_CNT_CLK)

- 64 bit gray code counter value generation
  - Convert the value generated by Timestamp generator to Gray code and output

- Access control
  - Access control for the two interfaces of the Timestamp generator
  - Control secure access/non-secure access

- Halt on Debug function
  - Counting can be stopped/restarted during debugging according to request from CoreSight

15.1.2 Block Diagram of SYC

The SYC connection diagram is shown in **Figure 15.1** and the SYC block diagram is shown in **Figure 15.2**.

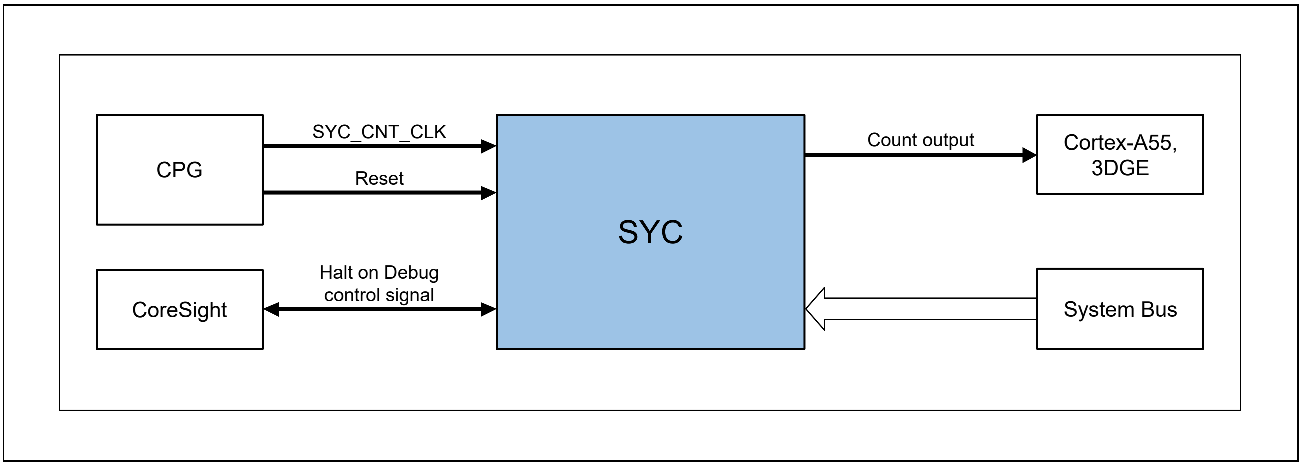


Figure 15.1 SYC Connection Diagram

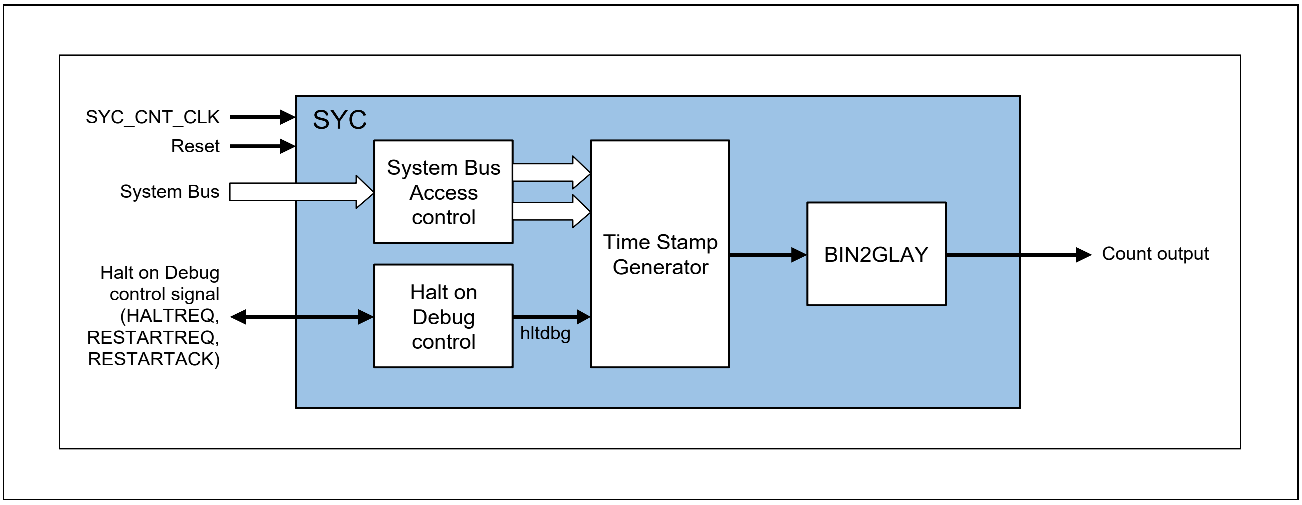


Figure 15.2 SYC Block Diagram

15.2 Address Space

Table 15.1 shows the SYC address space. SYC has an address space of 8 KB. The SYC address space is offset from the base address. The base address of SYC is as follows.

SYC base address: H’0_1100_0000 (Overall Address Space)
SYC base address: H’5100_0000 (Cortex-M33 Address Space Secure)
SYC base address: H’4100_0000 (Cortex-M33 Address Space Non-Secure)

Table 15.1 SYC Address Space

Offset Address	Region Name	Access Target
H’0000 to H’0FFF	PSELCTRL region	PSELCTRL region & PSELCTRL region Management registers
H’1000 to H’1FFF	PSELREAD region	PSELREAD region & PSELREAD region Management registers

Note: Secure access and non-secure access are possible for the PSELCTRL region and PSELREAD region.

15.3 Register Descriptions

For more information on SYC registers, please refer ARM CoreSight SoC-400 Technical Reference Manual 3.10 Timestamp generator.

15.4 Operation

15.4.1 System Bus Access Control

Access control from one system bus interface of SYC to two APB interfaces of Timestamp Generator is performed.

Controls so that the PSELCTRL area can be accessed when accessing the first half 4 KB of the 8 KB address area of SYC, and the PSELREAD area can be accessed when accessing the latter 4 KB.

The PSELCTRL area and PSELREAD area accept secure access and non-secure access.

15.4.2 Halt on Debug Function

SYC supports the Halt on Debug function included in Timestamp Generator. When HALTREQ is asserted from CoreSight, the counting operation of Timestamp Generator is stopped.

When RESTARTREQ is asserted from CoreSight, the counting operation of Timestamp Generator is restarted.

HALTREQ and RESTARTREQ are controlled by CoreSight's CTI (Cross Trigger Interface).

NOTE

This LSI supplies the common count value generated by SYC to Cortex-A55 and 3DGE. The stop/restart control of the count value by the Halt on Debug function is also performed for Cortex-A55 and 3DGE at the same time.

16. Multi-Function Timer Pulse Unit 3 (MTU3a)

16.1 Overview

This LSI has an on-chip multi-function timer pulse unit 3 (MTU3a), consisting of eight 16-bit timer channels and one 32- bit timer channel.

Table 16.1 shows the specifications of the MTU and **Table 16.2** lists the functions of the MTU. **Figure 16.1** and **Figure 16.2** are block diagrams of the MTU.

Table 16.1 MTU Specifications (1/2)

Item	Description
Pulse input/output	28 lines max.
Pulse input	3 lines
Count clock	11 clocks for each channel (14 clocks for MTU0, 12 clocks for MTU2, and 10 clocks for MTU5, four clocks for MTU1-MTU2 combination (when LWA = 1))
Operating frequency	Up to 100 MHz
Available operations	[MTU0 to MTU4, MTU6, MTU7, and MTU8] ● Waveform output on compare match ● Input capture function (noise filter setting available) ● Counter-clearing operation ● Simultaneous writing to multiple timer counters (TCNT) (excluding MTU8) ● Simultaneous clearing on compare match or input capture (excluding MTU8) ● Simultaneous input and output to registers in synchronization with counter operations (excluding MTU8) ● Up to 12-phase PWM output in combination with synchronous operation (excluding MTU8)
	[MTU0 MTU3, MTU4, MTU6, MTU7, and MTU8] ● Buffer operation specifiable
	[MTU1, MTU2] ● Phase counting mode can be specified independently ● 32-bit phase counting mode can be specified for interlocked operation of MTU1 and MTU2 (when TMDR3.LWA = 1) ● Cascade connection operation available
	[MTU3, MTU4, MTU6, and MTU7] ● Through interlocked operation of MTU3/4 and MTU6/7, the positive and negative signals in six phases (12 phases in total) can be output in complementary PWM and reset-synchronized PWM operation. ● In complementary PWM mode, values can be transferred from buffer registers to temporary registers at crests and troughs of the timer-counter values or when the buffer registers (TGRD registers in MTU4 and MTU7) are written to. ● Double-buffering selectable in complementary PWM mode
	[MTU3 and MTU4] ● Through interlocking with MTU0, a mode for driving AC synchronous motors (brushless DC motors) by using complementary PWM output and reset-synchronized PWM output is settable and allows the selection of two types of waveform output (chopping or level)
	[MTU5] ● Capable of operation as a dead-time compensation counter
	[MTU0/MTU5, MTU1, MTU2, and MTU8] 32-bit phase counting mode specifiable by combining MTU1 and MTU2 and through interlocked operation with MTU0/MTU5 and MTU8
Interrupt-skipping function	● In complementary PWM mode, interrupts on crests and troughs of counter values and triggers to start conversion by the A/D converter can be skipped
Interrupt sources	43 sources
Buffer operation	Automatic transfer of register data (transfer from the buffer register to the timer register)

Table 16.1 MTU Specifications (2/2)

Item	Description
Trigger generation	A/D converter start triggers can be generated
	A/D converter start request delaying function enables A/D converter to be started with any desired timing and to be synchronized with PWM output
Low power consumption function	The MTU3a can be placed in the module-stop state.

Table 16.2 MTU Functions (1/2)

Item		MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5	MTU6	MTU7	MTU8
Count clock		P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB MTCLKC MTCLKD MTIOC1A	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB MTCLKC	MTCLKA MTCLKB MTCLKC MTCLKD	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTIOC1A	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB	P0ϕ/1 P0ϕ/2 P0ϕ/4 P0ϕ/8 P0ϕ/16 P0ϕ/32 P0ϕ/64 P0ϕ/256 P0ϕ/1024 MTCLKA MTCLKB
External clocks in phase-counting mode		—	MTCLKA MTCLKB	MTCLKA MTCLKB MTCLKC MTCLKD	MTCLKA MTCLKB MTCLKC MTCLKD	—	—	—	—	—	—
General registers (TGR)		TGRA TGRB TGRE	TGRA TGRB	TGRA TGRB	TGRALW TGRBLW	TGRA TGRB	TGRA TGRB	TGRU TGRV TGRW	TGRA TGRB	TGRA TGRB	TGRA TGRB
General registers/ buffer registers		TGRC TGRD TGRF	—	—	—	TGRC TGRD TGRE	TGRC TGRD TGRE TGRF	—	TGRC TGRD TGRE	TGRC TGRD TGRE TGRF	TGRC TGRD
I/O pins		MTIOC0A MTIOC0B MTIOC0C MTIOC0D	MTIOC1A MTIOC1B	MTIOC2A MTIOC2B	MTIOC1A MTIOC1B	MTIOC3A MTIOC3B MTIOC3C MTIOC3D	MTIOC4A MTIOC4B MTIOC4C MTIOC4D	MTIC5U MTIC5V MTIC5W	MTIOC6A MTIOC6B MTIOC6C MTIOC6D	MTIOC7A MTIOC7B MTIOC7C MTIOC7D	MTIOC8A MTIOC8B MTIOC8C MTIOC8D
Counter clear function		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/ TGRBLW input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	✓	✓	✓	—	✓	✓	—	✓	✓	✓
	1 output	✓	✓	✓	—	✓	✓	—	✓	✓	✓
	Toggle output	✓	✓	✓	—	✓	✓	—	✓	✓	✓
Input capture function		✓	✓	✓	✓*1	✓	✓	✓	✓	✓	✓*2
Synchronous operation		✓	✓	✓	—	✓	✓	—	✓	✓	—
PWM mode 1		✓	✓	✓	—	✓	✓	—	✓	✓	—
PWM mode 2		✓	✓	✓	—	—	—	—	—	—	—
Complementary PWM mode		—	—	—	—	✓	✓	—	✓	✓	—
Reset-synchronized PWM mode		—	—	—	—	✓	✓	—	✓	✓	—
AC synchronous motor drive mode		✓	—	—	—	✓	✓	—	—	—	—
Phase counting mode		—	✓	✓	✓	—	—	—	—	—	—
Buffer operation		✓	—	—	—	✓	✓	—	✓	✓	✓
Dead time compensation counter function		—	—	—	—	—	—	—	—	—	—
DMAC activation		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGRALW/ TGRBLW input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow/ underflow*3	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow/ underflow*3	TGR compare match or input capture

Table 16.2 MTU Functions (2/2)

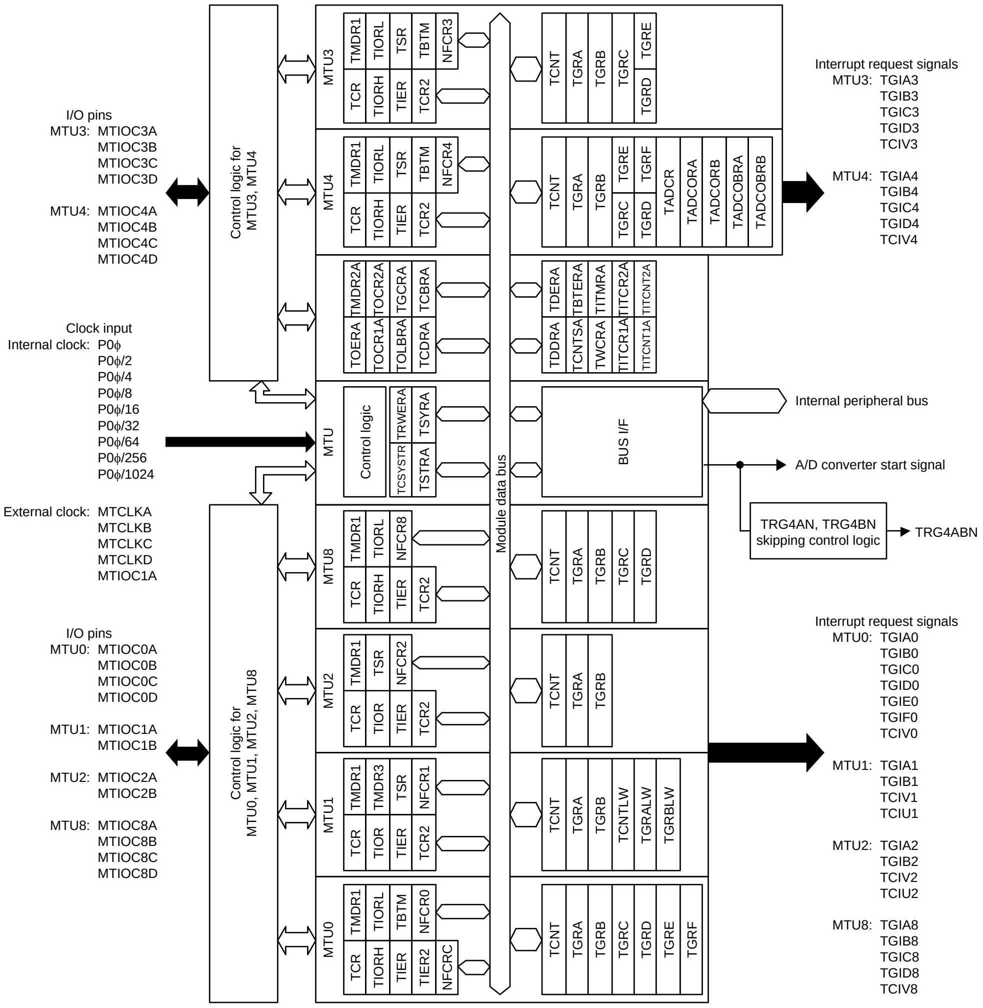
Item	MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU5	MTU6	MTU7	MTU8
A/D converter start trigger	TGRA compare match or input capture TGRE compare match	TGRA compare match or input capture	TGRA compare match or input capture	TGRALW input capture	TGRA compare match or input capture	TGRA compare match or input capture, or TCNT underflow (trough) in complementary PWM mode	—	TGRA compare match or input capture	TGRA compare match or input capture, or TCNT underflow (trough) in complementary PWM mode	—
Interrupt sources	Seven sources • Compare match or input capture 0A • Compare match or input capture 0B • Compare match or input capture 0C • Compare match or input capture 0D • Compare match 0E • Compare match 0F • Overflow	Four sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	Four sources • Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow	Four sources • Input capture 1A • Input capture 1B • Overflow • Underflow	Five sources • Compare match or input capture 3A • Compare match or input capture 3B • Compare match or input capture 3C • Compare match or input capture 3D • Overflow	Five sources • Compare match or input capture 4A • Compare match or input capture 4B • Compare match or input capture 4C • Compare match or input capture 4D • Overflow or underflow*3	Three sources • Compare match or input capture 5U • Compare match or input capture 5V • Compare match or input capture 5W	Five sources • Compare match or input capture 6A • Compare match or input capture 6B • Compare match or input capture 6C • Compare match or input capture 6D • Overflow	Five sources • Compare match or input capture 7A • Compare match or input capture 7B • Compare match or input capture 7C • Compare match or input capture 7D • Overflow or underflow*3	Five sources • Compare match or input capture 8A • Compare match or input capture 8B • Compare match or input capture 8C • Compare match or input capture 8D • Overflow
A/D converter start request delaying function	—	—	—	—	—	• A/D converter start request at a match between TADCORA and TCNT or A/D converter start request at a match between TADCORB and TCNT	—	—	• A/D converter start request at a match between TADCORA and TCNT or A/D converter start request at a match between TADCORB and TCNT	—
Interrupt skipping 1	—	—	—	—	• Skips TGRA compare match interrupts	• Skips TCIV interrupts	—	• Skips TGRA compare match interrupts	• Skips TCIV interrupts	—
Interrupt skipping 2	—	—	—	—	—	• Skipping in compare count between TADCORA and TCNT, and TADCORB and TCNT	—	—	• Skipping in compare count between TADCORA and TCNT, and TADCORB and TCNT	—

Remarks: ✓: Possible —: Not possible

Note 1. When LWA is 1, the TGRALW capture source can be selected from either of the following: an input from MTIOC1A or MTU0.TGRA compare match/input capture event.
The TGRBLW capture source can be selected from any of the following: an input from MTIOC1B, MTU0.TGRC compare match/ input capture event, or MTU8.TGRC compare match event.

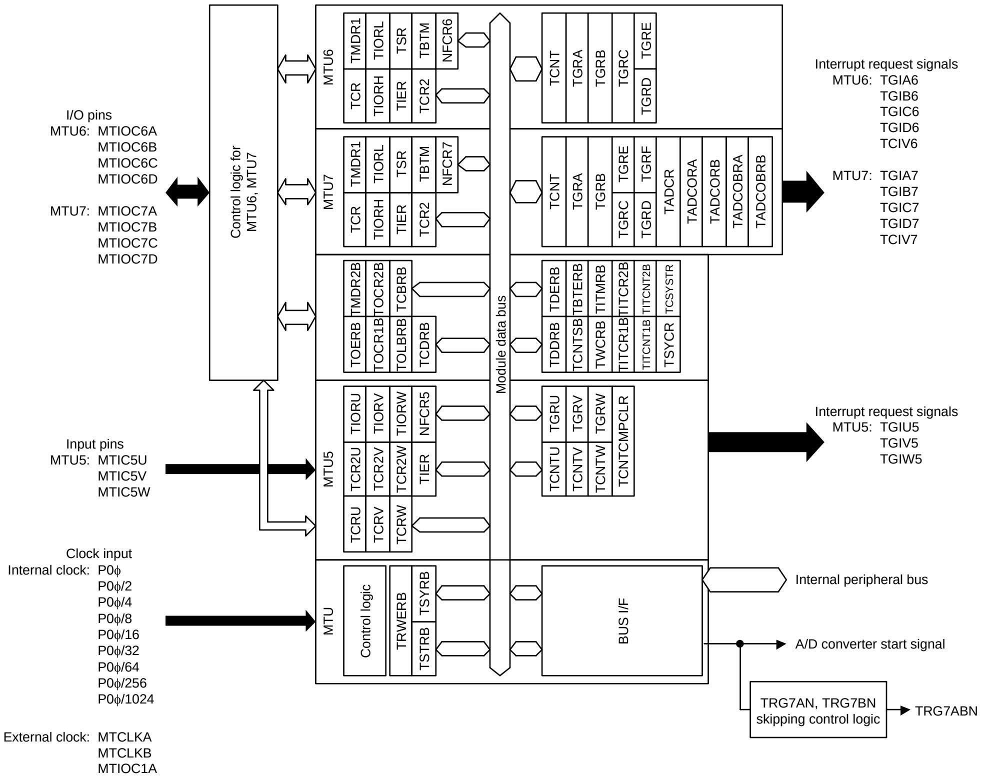
Note 2. Capture in MTU8 is supported only in normal mode.

Note 3. The underflow interrupt source is valid only in complementary PWM mode.



TCR:	Timer control register
TCR2:	Timer control register 2
TMDR:	Timer mode register 1
TMDR2A:	Timer mode register 2
TMDR3:	Timer mode register 3
TIOR:	Timer I/O control register
TIORH:	Timer I/O control register H
TIORL:	Timer I/O control register L
TIER:	Timer interrupt enable register
TIER2:	Timer interrupt enable register 2
TSR:	Timer status register
TBTM:	Timer buffer operation transfer mode register
TCNT:	Timer counter
TCNTLW:	Timer longword counter
TGRA:	Timer general register A
TGRALW:	Timer longword general register A
TGRB:	Timer general register B
TGRBLW:	Timer longword general register B
TGRC:	Timer general register C
TGRD:	Timer general register D
TGRE:	Timer general register E
TGRF:	Timer general register F
TSTRA:	Timer start register
TSYRA:	Timer synchronous register
TCSYSTR:	Timer counter synchronous start register
TRWERA:	Timer read/write enable register
TOERA:	Timer output master enable register
TOCR1A:	Timer output control register 1
TOCR2A:	Timer output control register 2
TOLBRA:	Timer output level buffer register
TGCRA:	Timer gate control register A
TCNTSA:	Timer subcounter
TCDRA:	Timer cycle data register
TCBRA:	Timer cycle buffer register
TDDRA:	Timer dead time data register
TDERA:	Timer dead time enable register
TBTERA:	Timer buffer transfer set register
TWCRA:	Timer waveform control register
TICCR:	Timer input capture control register
TADCR:	Timer A/D converter start request control register
TADCORA:	Timer A/D converter start request cycle set register A
TADCORB:	Timer A/D converter start request cycle set register B
TADCOBRA:	Timer A/D converter start request cycle set buffer register A
TADCOBRB:	Timer A/D converter start request cycle set buffer register B
TITMRA:	Timer interrupt skipping mode register
TITCR1A:	Timer interrupt skipping set register 1
TITCR2A:	Timer interrupt skipping set register 2
TITCNT1A:	Timer interrupt skipping counter 1
TITCNT2A:	Timer interrupt skipping counter 2
NFCRn:	Noise filter control register n

Figure 16.1 Block Diagram of MTU (MTU0 to MTU4, MTU8)



TCR: Timer control register
TCRU: Timer control register U
TCR2U: Timer control register 2U
TCRV: Timer control register V
TCR2V: Timer control register 2V
TCRW: Timer control register W
TCR2W: Timer control register 2W
TMDR1: Timer mode register 1
TMDR2B: Timer mode register 2
TIORH: Timer I/O control register H
TIORL: Timer I/O control register L
TIORU: Timer I/O control register U
TIORV: Timer I/O control register V
TIORW: Timer I/O control register W
TIER: Timer interrupt enable register
TSR: Timer status register
TCNT: Timer counter
TCNTU: Timer counter U
TCNTV: Timer counter V
TCNTW: Timer counter W
TGRA: Timer general register A
TGRB: Timer general register B
TGRC: Timer general register C
TGRD: Timer general register D
TGRE: Timer general register E
TGRF: Timer general register F
TGRU: Timer general register U
TGRV: Timer general register V
TGRW: Timer general register W
TCNTCMPCLR: Timer compare match clear register
TSTRB: Timer start register
TSYRB: Timer synchronous register
TSYCR: Timer synchronous clear register
TCSYSTR: Timer counter synchronous start register
TRWERB: Timer read/write enable register
TOERB: Timer output master enable register
TOCR1B: Timer output control register 1
TOCR2B: Timer output control register 2
TOLBRB: Timer output level buffer register
TCDRB: Timer cycle data register
TCBRB: Timer cycle buffer register
TDDRB: Timer dead time data register
TDERB: Timer dead time enable register
TCNTSB: Timer subcounter
TBTERB: Timer buffer transfer set register
TWCRB: Timer waveform control register
TADCR: Timer A/D converter start request control register
TADCORA: Timer A/D converter start request cycle set register A
TADCORB: Timer A/D converter start request cycle set register B
TADCOBRA: Timer A/D converter start request cycle set buffer register A
TADCOBRB: Timer A/D converter start request cycle set buffer register B
TITMRB: Timer interrupt skipping mode register
TITCR1B: Timer interrupt skipping set register 1
TITCR2B: Timer interrupt skipping set register 2
TITCNT1B: Timer interrupt skipping counter 1
TITCNT2B: Timer interrupt skipping counter 2
NFCRn Noise filter control register n

Figure 16.2 Block Diagram of MTU (MTU5 to MTU7)

Table 16.3 shows the configuration of pins for the MTU.

Table 16.3 Pin Configuration of the MTU

Channel	Pin Name	I/O	Function
MTU	MTCLKA	Input	External clock A input pin (MTU1/MTU2 phase counting mode A phase input)
	MTCLKB	Input	External clock B input pin (MTU1/MTU2 phase counting mode B phase input)
	MTCLKC	Input	External clock C input pin (MTU2 phase counting mode A phase input)
	MTCLKD	Input	External clock D input pin (MTU2 phase counting mode B phase input)
MTU0	MTIOC0A	I/O	MTU0 TGRA input capture input/output compare output/PWM output pin
	MTIOC0B	I/O	MTU0 TGRB input capture input/output compare output/PWM output pin
	MTIOC0C	I/O	MTU0 TGRC input capture input/output compare output/PWM output pin
	MTIOC0D	I/O	MTU0 TGRD input capture input/output compare output/PWM output pin
MTU1	MTIOC1A	I/O	MTU1 TGRA input capture input/output compare output/PWM output pin
	MTIOC1B	I/O	MTU1 TGRB input capture input/output compare output/PWM output pin
MTU2	MTIOC2A	I/O	MTU2 TGRA input capture input/output compare output/PWM output pin
	MTIOC2B	I/O	MTU2 TGRB input capture input/output compare output/PWM output pin
MTU3	MTIOC3A	I/O	MTU3 TGRA input capture input/output compare output/PWM output pin
	MTIOC3B	I/O	MTU3 TGRB input capture input/output compare output/PWM output pin
	MTIOC3C	I/O	MTU3 TGRC input capture input/output compare output/PWM output pin
	MTIOC3D	I/O	MTU3 TGRD input capture input/output compare output/PWM output pin
MTU4	MTIOC4A	I/O	MTU4 TGRA input capture input/output compare output/PWM output pin
	MTIOC4B	I/O	MTU4 TGRB input capture input/output compare output/PWM output pin
	MTIOC4C	I/O	MTU4 TGRC input capture input/output compare output/PWM output pin
	MTIOC4D	I/O	MTU4 TGRD input capture input/output compare output/PWM output pin
MTU5	MTIC5U	Input	MTU5 TGRU input capture input/external pulse input pin
	MTIC5V	Input	MTU5 TGRV input capture input/external pulse input pin
	MTIC5W	Input	MTU5 TGRW input capture input/external pulse input pin
MTU6	MTIOC6A	I/O	MTU6 TGRA input capture input/output compare output/PWM output pin
	MTIOC6B	I/O	MTU6 TGRB input capture input/output compare output/PWM output pin
	MTIOC6C	I/O	MTU6 TGRC input capture input/output compare output/PWM output pin
	MTIOC6D	I/O	MTU6 TGRD input capture input/output compare output/PWM output pin
MTU7	MTIOC7A	I/O	MTU7 TGRA input capture input/output compare output/PWM output pin
	MTIOC7B	I/O	MTU7 TGRB input capture input/output compare output/PWM output pin
	MTIOC7C	I/O	MTU7 TGRC input capture input/output compare output/PWM output pin
	MTIOC7D	I/O	MTU7 TGRD input capture input/output compare output/PWM output pin
MTU8	MTIOC8A	I/O	MTU8.TGRA input capture input/output compare output pin
	MTIOC8B	I/O	MTU8.TGRB input capture input/output compare output pin
	MTIOC8C	I/O	MTU8.TGRC input capture input/output compare output pin
	MTIOC8D	I/O	MTU8.TGRD input capture input/output compare output pin

16.2 Register Descriptions

Table 16.4 shows the register configuration. The MTU address space is offset from the base address. The base address of MTU is as follows.

MTU base address: H’0_1000_0000 (Overall Address Space)

MTU base address: H’5000_0000 (Cortex-M33 Address Space Secure)

MTU base address: H’4000_0000 (Cortex-M33 Address Space Non-Secure)

Table 16.4 Register configuration (1/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU0	Timer control register	TCR	H’1300	8
	Timer mode register 1	TMDR1	H’1301	8
	Timer I/O control register H	TIORH	H’1302	8
	Timer I/O control register L	TIORL	H’1303	8
	Timer interrupt enable register	TIER	H’1304	8
	Timer counter	TCNT	H’1306	16
	Timer general register A	TGRA	H’1308	16
	Timer general register B	TGRB	H’130A	16
	Timer general register C	TGRC	H’130C	16
	Timer general register D	TGRD	H’130E	16
	Timer general register E	TGRE	H’1320	16
	Timer general register F	TGRF	H’1322	16
	Timer interrupt enable register 2	TIER2	H’1324	8
	Timer buffer operation transfer mode register	TBTM	H’1326	8
	Timer control register 2	TCR2	H’1328	8
	Noise filter control register 0	NFCR0	H’1290	8
	Noise filter control register C	NFCRC	H’1299	8
MTU1	Timer control register	TCR	H’1380	8
	Timer mode register 1	TMDR1	H’1381	8
	Timer I/O control register	TIOR	H’1382	8
	Timer interrupt enable register	TIER	H’1384	8
	Timer status register	TSR	H’1385	8
	Timer counter	TCNT	H’1386	16
	Timer general register A	TGRA	H’1388	16
	Timer general register B	TGRB	H’138A	16
	Timer input capture control register	TICCR	H’1390	8
	Timer mode register 3	TMDR3	H’1391	8
	Timer control register 2	TCR2	H’1394	8
	Timer longword counter	TCNTLW	H’13A0	32
	Timer longword general register A	TGRALW	H’13A4	32
	Timer longword general register B	TGRBLW	H’13A8	32
	Noise filter control register 1	NFCR1	H’1291	8
MTU2	Timer control register	TCR	H’1400	8
	Timer mode register 1	TMDR1	H’1401	8
	Timer I/O control register	TIOR	H’1402	8
	Timer interrupt enable register	TIER	H’1404	8

Table 16.4 Register configuration (2/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU2	Timer status register	TSR	H'1405	8
	Timer counter	TCNT	H'1406	16
	Timer general register A	TGRA	H'1408	16
	Timer general register B	TGRB	H'140A	16
	Timer control register 2	TCR2	H'140C	8
	Noise filter control register 2	NFCR2	H'1292	8
MTU3	Timer control register	TCR	H'1200	8
	Timer mode register 1	TMDR1	H'1202	8
	Timer I/O control register H	TIORH	H'1204	8
	Timer I/O control register L	TIORL	H'1205	8
	Timer interrupt enable register	TIER	H'1208	8
	Timer counter	TCNT	H'1210	16
	Timer general register A	TGRA	H'1218	16
	Timer general register B	TGRB	H'121A	16
	Timer general register C	TGRC	H'1224	16
	Timer general register D	TGRD	H'1226	16
	Timer control register 2	TCR2	H'124C	8
	Timer general register E	TGRE	H'1272	16
	Timer status register	TSR	H'122C	8
	Timer buffer operation transfer mode register	TBTM	H'1238	8
	Noise filter control register 3	NFCR3	H'1293	8
MTU4	Timer control register	TCR	H'1201	8
	Timer mode register 1	TMDR1	H'1203	8
	Timer I/O control register H	TIORH	H'1206	8
	Timer I/O control register L	TIORL	H'1207	8
	Timer interrupt enable register	TIER	H'1209	8
	Timer counter	TCNT	H'1212	16
	Timer general register A	TGRA	H'121C	16
	Timer general register B	TGRB	H'121E	16
	Timer general register C	TGRC	H'1228	16
	Timer general register D	TGRD	H'122A	16
	Timer control register 2	TCR2	H'124D	8
	Timer general register E	TGRE	H'1274	16
	Timer general register F	TGRF	H'1276	16
	Timer status register	TSR	H'122D	8
	Timer buffer operation transfer mode register	TBTM	H'1239	8
	Timer A/D converter start request control register	TADCR	H'1240	16
	Timer A/D converter start request cycle set register A	TADCORA	H'1244	16
	Timer A/D converter start request cycle set register B	TADCORB	H'1246	16
	Timer A/D converter start request cycle set buffer register A	TADCOBRA	H'1248	16
	Timer A/D converter start request cycle set buffer register B	TADCOBRB	H'124A	16
	Noise filter control register 4	NFCR4	H'1294	8
MTU5	Timer counter U	TCNTU	H'1C80	16
	Timer general register U	TGRU	H'1C82	16
	Timer control register U	TCRU	H'1C84	8

Table 16.4 Register configuration (3/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU5	Timer control register 2U	TCR2U	H'1C85	8
	Timer I/O control register U	TIORU	H'1C86	8
	Timer counter V	TCNTV	H'1C90	16
	Timer general register V	TGRV	H'1C92	16
	Timer control register V	TCRV	H'1C94	8
	Timer control register 2V	TCR2V	H'1C95	8
	Timer I/O control register V	TIORV	H'1C96	8
	Timer counter W	TCNTW	H'1CA0	16
	Timer general register W	TGRW	H'1CA2	16
	Timer control register W	TCRW	H'1CA4	8
	Timer control register 2W	TCR2W	H'1CA5	8
	Timer I/O control register W	TIORW	H'1CA6	8
	Timer interrupt enable register	TIER	H'1CB2	8
	Timer start register	TSTR	H'1CB4	8
	Timer compare match clear register	TCNTCMPCLR	H'1CB6	8
	Noise filter control register 5	NFCR5	H'1A95	8
MTU6	Timer control register	TCR	H'1A00	8
	Timer mode register 1	TMDR1	H'1A02	8
	Timer I/O control register H	TIORH	H'1A04	8
	Timer I/O control register L	TIORL	H'1A05	8
	Timer interrupt enable register	TIER	H'1A08	8
	Timer counter	TCNT	H'1A10	16
	Timer general register A	TGRA	H'1A18	16
	Timer general register B	TGRB	H'1A1A	16
	Timer general register C	TGRC	H'1A24	16
	Timer general register D	TGRD	H'1A26	16
	Timer control register 2	TCR2	H'1A4C	8
	Timer general register E	TGRE	H'1A72	16
	Timer synchronous clear register	TSYCR	H'1A50	8
	Timer status register	TSR	H'1A2C	8
	Timer buffer operation transfer mode register	TBTM	H'1A38	8
	Noise filter control register 6	NFCR6	H'1A93	8
MTU7	Timer control register	TCR	H'1A01	8
	Timer mode register 1	TMDR1	H'1A03	8
	Timer I/O control register H	TIORH	H'1A06	8
	Timer I/O control register L	TIORL	H'1A07	8
	Timer interrupt enable register	TIER	H'1A09	8
	Timer counter	TCNT	H'1A12	16
	Timer general register A	TGRA	H'1A1C	16
	Timer general register B	TGRB	H'1A1E	16
	Timer general register C	TGRC	H'1A28	16
	Timer general register D	TGRD	H'1A2A	16
	Timer control register 2	TCR2	H'1A4D	8
	Timer general register E	TGRE	H'1A74	16
	Timer general register F	TGRF	H'1A76	16

Table 16.4 Register configuration (4/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU7	Timer status register	TSR	H'1A2D	8
	Timer buffer operation transfer mode register	TBTM	H'1A39	8
	Timer A/D converter start request control register	TADCR	H'1A40	16
	Timer A/D converter start request cycle set register A	TADCORA	H'1A44	16
	Timer A/D converter start request cycle set register B	TADCORB	H'1A46	16
	Timer A/D converter start request cycle set buffer register A	TADCOBRA	H'1A48	16
	Timer A/D converter start request cycle set buffer register B	TADCOBRB	H'1A4A	16
	Noise filter control register 7	NFCR7	H'1A94	8
MTU8	Timer control register	TCR	H'1600	8
	Timer mode register 1	TMDR1	H'1601	8
	Timer I/O control register H	TIORH	H'1602	8
	Timer I/O control register L	TIORL	H'1603	8
	Timer interrupt enable register	TIER	H'1604	8
	Timer control register 2	TCR2	H'1606	8
	Timer counter	TCNT	H'1608	32
	Timer general register A	TGRA	H'160C	32
	Timer general register B	TGRB	H'1610	32
	Timer general register C	TGRC	H'1614	32
	Timer general register D	TGRD	H'1618	32
	Noise filter control register 8	NFCR8	H'1298	8
MTU	Timer output master enable register A	TOERA	H'120A	8
	Timer gate control register A	TGCRA	H'120D	8
	Timer output control register 1A	TOCR1A	H'120E	8
	Timer output control register 2A	TOCR2A	H'120F	8
	Timer cycle data register A	TCDRA	H'1214	16
	Timer dead time data register A	TDDRA	H'1216	16
	Timer subcounter A	TCNTSA	H'1220	16
	Timer cycle buffer register A	TCBRA	H'1222	16
	Timer interrupt skipping set register 1A	TITCR1A	H'1230	8
	Timer interrupt skipping counter 1A	TITCNT1A	H'1231	8
	Timer buffer transfer set register A	TBTERA	H'1232	8
	Timer dead time enable register A	TDERA	H'1234	8
	Timer output level buffer register A	TOLBRA	H'1236	8
	Timer interrupt skipping mode register A	TITMRA	H'123A	8
	Timer interrupt skipping set register 2A	TITCR2A	H'123B	8
	Timer interrupt skipping counter 2A	TITCNT2A	H'123C	8
	Timer waveform control register A	TWCRA	H'1260	8
	Timer mode register 2A	TMDR2A	H'1270	8
	Timer read/write enable register A	TRWERA	H'1284	8
	Timer start register A	TSTRA	H'1280	8
	Timer synchronous register A	TSYRA	H'1281	8
	Timer output master enable register B	TOERB	H'1A0A	8
	Timer output control register 1B	TOCR1B	H'1A0E	8
	Timer output control register 2B	TOCR2B	H'1A0F	8
	Timer cycle data register B	TCDRB	H'1A14	16

Table 16.4 Register configuration (5/5)

Channel	Register Name	Abbreviation	Address	Access size
MTU	Timer dead time data register B	TDDRB	H’1A16	16
	Timer subcounter B	TCNTSB	H’1A20	16
	Timer cycle buffer register B	TCBRB	H’1A22	16
	Timer interrupt skipping set register 1B	TITCR1B	H’1A30	8
	Timer interrupt skipping counter 1B	TITCNT1B	H’1A31	8
	Timer buffer transfer set register B	TBTERB	H’1A32	8
	Timer dead time enable register B	TDERB	H’1A34	8
	Timer output level buffer register B	TOLBRB	H’1A36	8
	Timer interrupt skipping mode register B	TITMRB	H’1A3A	8
	Timer interrupt skipping set register 2B	TITCR2B	H’1A3B	8
	Timer interrupt skipping counter 2B	TITCNT2B	H’1A3C	8
	Timer waveform control register B	TWCRB	H’1A60	8
	Timer mode register 2B	TMDR2B	H’1A70	8
	Timer start register B	TSTRB	H’1A80	8
	Timer synchronous register B	TSYRB	H’1A81	8
	Timer read/write enable register B	TRWERB	H’1A84	8
	Timer counter synchronous start register	TCSYSTR	H’1282	8

16.2.1 Timer Control Register (TCR)

(1) MTU0.TCR, MTU1.TCR, MTU2.TCR, MTU3.TCR, MTU4.TCR, MTU6.TCR, MTU7.TCR, MTU8.TCR

Address(es): MTU0.TCR H’0_1000_1300, MTU1.TCR H’0_1000_1380, MTU2.TCR H’0_1000_1400, MTU3.TCR H’0_1000_1200, MTU4.TCR H’0_1000_1201, MTU6.TCR H’0_1000_1A00, MTU7.TCR H’0_1000_1A01, MTU8.TCR H’0_1000_1600

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TPSC[2:0]	All 0	R/W	Time Prescaler Select See **Table 16.7** to **Table 16.10**.
b4, b3	CKEG[1:0]	All 0	R/W	Clock Edge Select b4 b3 0 0: Count at rising edge 0 1: Count at falling edge 1 x: Count at both edges
b7 to b5	CCLR[2:0]	All 0	R/W	Counter Clear Source Select See **Table 16.5** and **Table 16.6**.

Remarks: x: Don’t care

The TCR register controls the TCNT operation for each channel in combination with the TCR2 register. The MTU has a total of 11 TCR registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8 and three (TCRU, TCRV, and TCRW) for MTU5. TCR values should be specified only while TCNT operation is stopped.

TPSC[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. The clock source can be selected independently for each channel. See **Table 16.7** to **Table 16.10** for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the count clock source edge, including the MTIOC1A pin. When the internal clock is counted at both edges, the count clock period is halved (e.g. P0ϕ/4 at both edges = P0ϕ/2 at rising edge). If phase counting mode is used on MTU1 and MTU2, the setting of these bits is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the count clock source is P0ϕ/2 or slower. When P0ϕ/1 or the overflow/underflow in another channel is selected for the count clock source, a value can be written to these bits but counter operation compiles with the initial value.

CCLR[2:0] Bits (Counter Clear Source Select)

These bits select the TCNT counter clearing source. See **Table 16.5** and **Table 16.6** for details.

Table 16.5 CCLR[2:0] (MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8)

	Bit 7	Bit 6	Bit 5	
Channel	CCLR2	CCLR1	CCLR0	Description
MTU0 MTU3 MTU4 MTU6 MTU7 MTU8	0	0	0	TCNT clearing disabled
	0	0	1	TCNT cleared by TGRA compare match/input capture
	0	1	0	TCNT cleared by TGRB compare match/input capture
	0	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*[1]
	1	0	0	TCNT clearing disabled
	1	0	1	TCNT cleared by TGRC compare match/input capture*[2]
	1	1	0	TCNT cleared by TGRD compare match/input capture*[2]
	1	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*[1]

Note 1. Synchronous operation is selected by setting a SYNC bit in TSYRA or TSYRB to 1 except for MTU8.

Note 2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority and compare match/input capture does not occur.

Table 16.6 CCLR[2:0] (MTU1 and MTU2)

	Bit 7	Bit 6	Bit 5	
Channel	Reserved*[2]	CCLR1	CCLR0	Description
MTU1 MTU2	0	0	0	TCNT clearing disabled
	0	0	1	TCNT cleared by TGRA compare match/input capture
	0	1	0	TCNT cleared by TGRB compare match/input capture
	0	1	1	TCNT cleared by counter clearing in another channel where synchronous clearing or synchronous operation is selected*[1]

Note 1. Synchronous operation is selected by setting a SYNC bit in TSYRA or TSYRB to 1.

Note 2. Bit 7 is reserved in MTU1 and MTU2. It is read as 0. The write value is ignored.

(2) MTU5.TCRU, MTU5.TCRV, MTU5.TCRW

Address(es): MTU5.TCRU H'0_1000_1C84, MTU5.TCRV H'0_1000_1C94, MTU5.TCRW H'0_1000_1CA4

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TPSC[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	TPSC[1:0]	All 0	R/W	Time Prescaler Select See **Table 16.11**.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TPSC[1:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. See **Table 16.11** for details.

16.2.2 Timer Control Register 2 (TCR2)

(1) MTU0.TCR2, MTU3.TCR2, MTU4.TCR2, MTU6.TCR2, MTU7.TCR2, MTU8.TCR2

Address(es): MTU0.TCR2 H'0_1000_1328, MTU3.TCR2 H'0_1000_124C, MTU4.TCR2 H'0_1000_124D,
MTU6.TCR2 H'0_1000_1A4C, MTU7.TCR2 H'0_1000_1A4D, MTU8.TCR2 H'0_1000_1606

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TPSC2[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU1.TCR2, MTU2.TCR2

Address(es): MTU1.TCR2 H'0_1000_1394, MTU2.TCR2 H'0_1000_140C

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PCB[1:0]		TPSC2[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TPSC2[2:0]	All 0	R/W	Time Prescaler Select See **Table 16.7** to **Table 16.10**.
b4, b3	PCB[1:0]	All 0	R/W	Phase Counting Mode Function Expansion Control Functional Expansion Control for Phase Counting Modes 2, 3, and 5
b7 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The TCR2 register controls the TCNT operation for each channel in combination with the TCR register. The MTU has a total of 11 TCR2 registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8 and three (TCR2U, TCR2V, and TCR2W) for MTU5. TCR2 values should be specified only while TCNT operation is stopped.

TPSC2[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. The clock source can be selected independently for each channel. See **Table 16.7** to **Table 16.10** for details.

PCB[1:0] Bits (Phase Counting Mode Function Expansion Control)

These bits control extended functions for phase counting mode 2, 3, and 5 in MTU1 and MTU2. See **Section 16.3.6, Phase Counting Mode**.

(3) MTU5.TCR2U, MTU5.TCR2V, MTU5.TCR2W

Address(es): MTU5.TCR2U H'0_1000_1C85, MTU5.TCR2V H'0_1000_1C95, MTU5.TCR2W H'0_1000_1CA5

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	CKEG[1:0]		TPSC2[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TPSC2[2:0]	All 0	R/W	Time Prescaler Select See **Table 16.11**.
b4, b3	CKEG[1:0]	All 0	R/W	Clock Edge Select b4 b3 0 0: Counts at the rising edge. 0 1: Counts at the falling edge. 1 x: Counts at both edges.
b7 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Remarks: x: Don’t care

TPSC2[2:0] Bits (Time Prescaler Select)

These bits select the TCNT count clock source. See **Table 16.11** for details.

CKEG[1:0] Bits (Clock Edge Select)

These bits select the edge of the count clock source signal output from the MTIOC1A pin.

Table 16.7 TPSC[2:0], TPSC2[2:0] (MTU0)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU0	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	External clock: counts on MTCLKD pin input
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Internal clock: counts on P0ϕ/256
	1	0	1	x	x	x	Internal clock: counts on P0ϕ/1024
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	External clock: counts on MTIOC1A pin input

Remarks: x: Don’t care

Table 16.8 TPSC[2:0], TPSC2[2:0] (MTU1)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU1	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	Internal clock: counts on P0ϕ/256
	0	0	0	1	1	1	Overflow/underflow of MTU2.TCNT
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Internal clock: counts on P0ϕ/1024
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

Note: This setting has no effect when MTU1 is in phase counting mode.

Remarks: x: Don’t care

Table 16.9 TPSC[2:0], TPSC2[2:0] (MTU2)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU2	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	External clock: counts on MTCLKA pin input
	0	0	0	1	0	1	External clock: counts on MTCLKB pin input
	0	0	0	1	1	0	External clock: counts on MTCLKC pin input
	0	0	0	1	1	1	Internal clock: counts on P0ϕ/1024
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Internal clock: counts on P0ϕ/256
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

Note: This setting has no effect when the MTU2 is in phase counting mode.

Remarks: x: Don't care

Table 16.10 TPSC[2:0], TPSC2[2:0] (MTU3, MTU4, MTU6, MTU7, and MTU8)

	TCR2[2:0]			TCR[2:0]			
	Bit 2	Bit 1	Bit 0	Bit 2	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC2	TPSC1	TPSC0	Description
MTU3 MTU4 MTU6 MTU7 MTU8	0	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	0	1	0	0	Internal clock: counts on P0ϕ/256
	0	0	0	1	0	1	Internal clock: counts on P0ϕ/1024
	0	0	0	1	1	0	External clock: counts on MTCLKA pin input
	0	0	0	1	1	1	External clock: counts on MTCLKB pin input
	0	0	1	x	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	x	Setting prohibited
	1	0	1	x	x	x	Setting prohibited
	1	1	0	x	x	x	Setting prohibited
	1	1	1	x	x	x	Setting prohibited

Remarks: x: Don't care

Table 16.11 TPSC[1:0], TPSC2[2:0] (MTU5)

	TCR2[2:0]			TCR[1:0]		
	Bit 2	Bit 1	Bit 0	Bit 1	Bit 0	
Channel	TPSC22	TPSC21	TPSC20	TPSC1	TPSC0	Description
MTU5	0	0	0	0	0	Internal clock: counts on P0ϕ/1
	0	0	0	0	1	Internal clock: counts on P0ϕ/4
	0	0	0	1	0	Internal clock: counts on P0ϕ/16
	0	0	0	1	1	Internal clock: counts on P0ϕ/64
	0	0	1	x	x	Internal clock: counts on P0ϕ/2
	0	1	0	x	x	Internal clock: counts on P0ϕ/8
	0	1	1	x	x	Internal clock: counts on P0ϕ/32
	1	0	0	x	x	Internal clock: counts on P0ϕ/256
	1	0	1	x	x	Internal clock: counts on P0ϕ/1024
	1	1	0	x	x	Setting prohibited
	1	1	1	x	x	Internal clock: counts on MTIOC1A pin input

Remarks: x: Don’t care

16.2.3 Timer Mode Register 1 (TMDR1)

(1) MTU0.TMDR1

Address(es): MTU0.TMDR1 H'0_1000_1301

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	BFE	BFB	BFA	MD[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU1.TMDR1, MTU2.TMDR1

Address(es): MTU1.TMDR1 H'0_1000_1381, MTU2.TMDR1 H'0_1000_1401

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MD[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(3) MTU3.TMDR1, MTU4.TMDR1, MTU6.TMDR1, MTU7.TMDR1, MTU8.TMDR1

Address(es): MTU3.TMDR1 H'0_1000_1202, MTU4.TMDR1 H'0_1000_1203, MTU6.TMDR1 H'0_1000_1A02,
MTU7.TMDR1 H'0_1000_1A03, MTU8.TMDR1 H'0_1000_1601

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	BFB	BFA	MD[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b3 to b0	MD[3:0]	All 0	R/W	Mode Select These bits specify the timer operating mode. See **Table 16.12** for details.
b4	BFA	0	R/W	Buffer Operation A 0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation
b5	BFB	0	R/W	Buffer Operation B 0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation
b6	BFE	0	R/W	Buffer Operation E 0: MTU0.TGRE and MTU0.TGRF operate normally 1: MTU0.TGRE and MTU0.TGRF used together for buffer operation
b7	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

The TMDR1 register specifies the operating mode of each channel. The MTU has a total of eight TMDR1 registers, one each for MTU0 to MTU4, MTU 6, MTU7, and MTU8. TMDR1 register values should be specified only while TCNT operation is stopped.

Table 16.12 Operating Mode Setting by MD[3:0] Bits (MTU0 to MTU4, MTU6 to MTU8)

Bit 3	Bit 2	Bit 1	Bit 0		MTU0	MTU1	MTU2	MTU1 & MTU2 (LWA = 1)	MTU3	MTU4	MTU6	MTU7	MTU8
MD3	MD2	MD1	MD0	Description									
0	0	0	0	Normal mode	✓	✓	✓		✓	✓	✓	✓	✓
0	0	0	1	Setting prohibited									
0	0	1	0	PWM mode 1	✓	✓	✓		✓	✓	✓	✓	
0	0	1	1	PWM mode 2	✓	✓	✓						
0	1	0	0	Phase counting mode 1		✓	✓	✓					
0	1	0	1	Phase counting mode 2		✓	✓	✓					
0	1	1	0	Phase counting mode 3		✓	✓	✓					
0	1	1	1	Phase counting mode 4		✓	✓	✓					
1	0	0	0	Reset-synchronized PWM mode*[1]					✓		✓		
1	0	0	1	Phase counting mode 5		✓	✓	✓					
1	0	1	x	Setting prohibited									
1	1	0	0	Setting prohibited									
1	1	0	1	Complementary PWM mode 1 (transfer at crest)*[1]					✓		✓		
1	1	1	0	Complementary PWM mode 2 (transfer at trough)*[1]					✓		✓		
1	1	1	1	Complementary PWM mode 3 (transfer at crest and trough)*[1]					✓		✓		

Note: Only set the corresponding operating mode listed above for each channel.

Remarks: x: Don't care

Note 1. Reset-synchronized PWM mode and complementary PWM mode can only be set for MTU3 and MTU6. When MTU3 or MTU6 is set to reset-synchronized PWM mode or complementary PWM mode, the MTU4 or MTU7 settings become ineffective and automatically conform to the MTU3 or MTU6 setting, respectively. MTU4 and MTU7 should be set to the initial values (normal mode).

BFA Bit (Buffer Operation A)

This bit specifies whether to operate TGRA in the normal way or to use TGRA and TGRC together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRC occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIEC bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in the BFA bit of MTU3.TMDR1 (MTU6.TMDR1). The BFA bit of MTU4.TMDR1 (MTU7.TMDR1) should be set to 0.

In MTU1 and MTU2, which have no TGRC, this bit is reserved. It is read as 0. The write value should be 0. See **Figure 16.50** for an illustration of the Tb interval in complementary PWM mode.

BFB Bit (Buffer Operation B)

This bit specifies whether to operate TGRB in the normal way or to use TGRB and TGRD together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare does not take place in modes other than complementary PWM mode, but compare match with TGRD occurs in complementary PWM mode.

If a compare match occurs on MTU4 in the Tb interval in complementary PWM mode, the TGIED bit in timer interrupt enable register (MTU4.TIER) should be set to 0.

In reset-synchronized PWM mode or complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in the BFB bit of MTU3.TMDR1 (MTU6.TMDR1). The BFB bit of MTU4.TMDR1 (MTU7.TMDR1) should be set to 0.

In MTU1 and MTU2, which have no TGRD, this bit is reserved. It is read as 0. The write value should be 0. See **Figure 16.50** for an illustration of the Tb interval in complementary PWM mode.

BFE Bit (Buffer Operation E)

This bit specifies whether to operate MTU0.TGRE and MTU0.TGRF in the normal way or to use them together for buffer operation. Compare match with TGRF occurs even when TGRF is used as a buffer register.

In MTU0 to MTU4, MTU6, MTU7, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

16.2.4 Timer Mode Registers 2 (TMDR2A and TMDR2B)

Address(es): MTU.TMDR2A H'0_1000_1270, MTU.TMDR2B H'0_1000_1A70

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	DRS
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	DRS	0	R/W	Double Buffer Select 0: Double buffer function is disabled 1: Double buffer function is enabled
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TMDR2A and TMDR2B specify the double buffer function in complementary PWM mode 3 (transfer at the crest and trough of the counter value). The MTU has two TMDR2 registers, one each for MTU3 (TMDR2A) and MTU6 (TMDR2B). TMDR2A and TMDR2B values should be specified only while TCNT operation is stopped.

DRS Bit (Double Buffer Select)

This bit enables or disables the double buffer function in complementary PWM mode.

16.2.5 Timer Mode Register 3 (TMDR3)

Address(es): MTU1.TMDR3 H'0_1000_1391

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PHCKSEL	LWA
Initial Value	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	LWA	0	R/W	MTU1/MTU2 Combination Longword Access Control 0: 16-bit access is enabled. 1: 32-bit access is enabled.
b1	PHCKSEL	1	R/W	External Input Phase Clock Select Selects the external clock pin for phase counting mode. 0: MTCLKA and MTCLKB are selected for the external phase clock. 1: MTCLKC and MTCLKD are selected for the external phase clock.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The TMDR3 register controls longword access to a 32-bit register or counter in a combination of MTU1 and MTU2. There is only one TMDR3 register in MTU1. The counter (TCNTLW), general register A (TGRALW), and general register B (TGRBLW) of MTU1 and MTU2 are accessed in the combinations listed in **Table 16.13**.

LWA Bit (MTU1/MTU2 Combination Longword Access Control)

This bit selects a 32-bit access in a combination of MTU1 and MTU2.

When the LWA bit is set to 0, MTU1 and MTU2 operate independently as 16-bit timers and the TCNTLW, TGRALW, and TGRBLW registers cannot be accessed.

When the LWA bit is set to 1, MTU1 and MTU2 are cascaded and operate as a 32-bit timer, which is controlled through the TCR, TCR2, TIOR, and TMDR1 registers in MTU1. The TCR, TCR2, TIOR, and TMDR1 register settings in MTU2 are ignored. The 16-bit registers (TCNT, TGRA, and TGRB) in MTU1 and MTU2 cannot be accessed. MTU2 input capture and compare match are disabled.

Note that MTU1 and MTU2 can be cascaded by setting the LWA bit to 1 only in phase-counting mode. Cascade connection cannot be used in normal, PWM1, or PWM2 mode. When setting the LWA bit to 1, be sure to select phase-counting mode.

In addition, initialize the TCNT, TGRA, and TGRB registers in MTU1 and MTU2 before setting the LWA bit to 1.

PHCKSEL Bit (External Input Phase Clock Select)

When the MTU1 and MTU2 registers are combined for 32-bit phase counting mode or MTU2 phase counting mode, this bit selects either the A- or B-phase signal from the external clock. See **Table 16.66, Clock Input Pins in Phase Counting Mode** for details.

Table 16.13 Setting and Combination of the TMDR3 Register

	TMDR3.LWA = 0		TMDR3.LWA = 1	
Register	Symbol	Access mode	Symbol	Access mode
Counter in MTU1*[1]	MTU1.TCNT	Word	MTU1.TCNT_1_LW	Longword
Counter in MTU2	MTU2.TCNT	Word		
General register A in MTU1	MTU1.TGRA	Word	MTU1.TGRA_1_LW	Longword
General register A in MTU2	MTU2.TGRA	Word		
General register B in MTU1	MTU1.TGRB	Word	MTU1.TGRB_1_LW	Longword
General register B in MTU2	MTU2.TGRB	Word		

Note 1. When the LWA bit is set to 1, setting the count clock for MTU1 as MTU2.TCNT overflow/underflow is not required.

16.2.6 Timer I/O Control Register (TIOR)

(1) MTU0.TIORH, MTU1.TIOR, MTU2.TIOR, MTU3.TIORH, MTU4.TIORH, MTU6.TIORH, MTU7.TIORH, MTU8.TIORH

Address(es): MTU0.TIORH H’0_1000_1302, MTU1.TIOR H’0_1000_1382, MTU2.TIOR H’0_1000_1402, MTU3.TIORH H’0_1000_1204, MTU4.TIORH H’0_1000_1206, MTU6.TIORH H’0_1000_1A04, MTU7.TIORH H’0_1000_1A06, MTU8.TIORH H’0_1000_1602

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	IOB[3:0]				IOA[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b3 to b0	IOA[3:0]	All 0	R/W	I/O Control A*[1] See the following tables. MTU0.TIORH: **Table 16.28** MTU1.TIOR: **Table 16.30** MTU2.TIOR: **Table 16.31** MTU3.TIORH: **Table 16.32** MTU4.TIORH: **Table 16.34** MTU6.TIORH: **Table 16.36** MTU7.TIORH: **Table 16.38** MTU8.TIORH: **Table 16.40**
b7 to b4	IOB[3:0]	All 0	R/W	I/O Control B*[1] See the following tables. MTU0.TIORH: **Table 16.14** MTU1.TIOR: **Table 16.16** MTU2.TIOR: **Table 16.17** MTU3.TIORH: **Table 16.18** MTU4.TIORH: **Table 16.20** MTU6.TIORH: **Table 16.22** MTU7.TIORH: **Table 16.24** MTU8.TIORH: **Table 16.26**

Note 1. When the value of IOn[3:0] (n = A, B) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

(2) MTU0.TIORL, MTU3.TIORL, MTU4.TIORL, MTU6.TIORL, MTU7.TIORL, MTU8.TIORL

Address(es): MTU0.TIORL H’0_1000_1303, MTU3.TIORL H’0_1000_1205, MTU4.TIORL H’0_1000_1207, MTU6.TIORL H’0_1000_1A05,MTU7.TIORL H’0_1000_1A07, MTU8.TIORL H’0_1000_1603

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	IOD[3:0]				IOC[3:0]			
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b3 to b0	IOC[3:0]	All 0	R/W	I/O Control C*[1] See the following tables. MTU0.TIORL: **Table 16.29** MTU3.TIORL: **Table 16.33** MTU4.TIORL: **Table 16.35** MTU6.TIORL: **Table 16.37** MTU7.TIORL: **Table 16.39** MTU8.TIORL: **Table 16.41**
b7 to b4	IOD[3:0]	All 0	R/W	I/O Control D*[1] See the following tables. MTU0.TIORL: **Table 16.15** MTU3.TIORL: **Table 16.19** MTU4.TIORL: **Table 16.21** MTU6.TIORL: **Table 16.23** MTU7.TIORL: **Table 16.25** MTU8.TIORL: **Table 16.27**

Note 1. When the value of IOn[3:0] (n = C, D) is changed to the output-prohibited state (0000b or 0100b) during low, high, or toggle output at compare match, this register is in Hi-Z.

(3) MTU5.TIORU, MTU5.TIORV, MTU5.TIORW

Address(es): MTU5.TIORU H’0_1000_1C86, MTU5.TIORV H’0_1000_1C96, MTU5.TIORW H’0_1000_1CA6

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IOC[4:0]				
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	IOC[4:0]	All 0	R/W	I/O Control C See the following table. MTU5.TIORU, MTU5.TIORV, MTU5.TIORW: **Table 16.42**
b7 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The TIOR register controls the TGR register. The MTU has a total of 17 TIOR registers, two each for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, one each for MTU1 and MTU2, and three (MTU5.TIORU/TIORV/TIORW) for MTU5. The TIOR register should be set when the TMDR register setting is normal mode, PWM mode, or phase counting mode.

Note that TIOR is affected by the TMDR1 setting.

The initial output specified by TIOR is valid when the counter is stopped (the CSTn bits in TSTRA and TSTRB are cleared to 0). Note also that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified. When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

Table 16.14 TIORH (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU0.TGRB Function	MTIOC0B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*[1]

Remarks: Don’t care

Note 1. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.15 TIORL (MTU0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU0.TGRD Function	MTIOC0D Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*[1]	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*[2]

Remarks: Don't care

Note 1. When the MTU0.TMDR1.BFB is set to 1 and MTU0.TGRD is used as a buffer register, this setting is invalid and input capture/ output compare is not generated.

Note 2. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.16 TIOR (MTU1)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU1.TGRB/TGRBLW Function	MTIOC1B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	0	0		Input capture at occurrence of compare match or input capture in the MTU0.TGRC register
1	1	1	x		Input capture at occurrence of compare match in the MTU8.TGRC register

Remarks: x: Don’t care

Table 16.17 TIOR (MTU2)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU2.TGRB Function	MTIOC2B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.18 TIORH (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU3.TGRB Function	MTIOC3B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.19 TIORL (MTU3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU3.TGRD Function	MTIOC3D Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*[1]	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU3.TMDR1 is set to 1 and MTU3.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.20 TIORH (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU4.TGRB Function	MTIOC4B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.21 TIORL (MTU4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU4.TGRD Function	MTIOC4D Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*[1]	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU4.TMDR1 is set to 1 and MTU4.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.22 TIORH (MTU6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU6.TGRB Function	MTIOC6B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.23 TIORL (MTU6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU6.TGRD Function	MTIOC6D Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*[1]	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU6.TMDR1 is set to 1 and MTU6.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.24 TIORH (MTU7)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU7.TGRB Function	MTIOC7B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.25 TIORL (MTU7)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU7.TGRD Function	MTIOC7D Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*[1]	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFB bit in MTU7.TMDR1 is set to 1 and MTU7.TGRD register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.26 TIORH (MTU8)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MTU8.TGRB Function	MTIOC8B Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by MTU1.TCNT (LWA = 0) or MTU1.TCNTLW (LWA = 1)*[1]

Remarks: x: Don't care

Note 1. Input capture will not be generated in MTU8 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.27 TIORL (MTU8)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MTU8.TGRD Function	MTIOC8D Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Note 1. When the BFB bit of the TMDR1 register is set to 1 and the TGRD register is used as a buffer register in MTU8, this setting is invalid and input capture/output compare is not generated.

Table 16.28 TIORH (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU0.TGRA Function	MTIOC0A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	0	0		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*[1]
1	1	1	x		Input capture on generation of compare match with MTU8.TGRC

Remarks: x: Don’t care

Note 1. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.29 TIORL (MTU0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU0.TGRC Function	MTIOC0C Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register*[1]	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Capture input source is the clock source for counting in MTU1. Input capture on counting up or down by TCNT (LWA = 0) or TCNTLW (LWA = 1) in MTU1*[2]

Remarks: x: Don’t care

Note 1. When the BFA bit in MTU0.TMDR1 is set to 1 and MTU0.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Note 2. Input capture will not be generated in MTU0 if P0ϕ/1 is selected as the count clock for MTU1. Select a clock other than P0ϕ/1.

Table 16.30 TIOR (MTU1)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU1.TGRA/TGRALW Function	MTIOC1A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	0	1	x		Input capture at both edges.
1	1	x	x		Input capture at generation of MTU0.TGRA compare match/input capture.

Remarks: x: Don’t care

Table 16.31 TIOR (MTU2)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU2.TGRA Function	MTIOC2A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.32 TIORH (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU3.TGRA Function	MTIOC3A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.33 TIORL (MTU3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU3.TGRC Function	MTIOC3C Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*[1]	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFA bit in MTU3.TMDR1 is set to 1 and MTU3.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.34 TIORH (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU4.TGRA Function	MTIOC4A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Table 16.35 TIORL (MTU4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU4.TGRC Function	MTIOC4C Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*[1]	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Note 1. When the BFA bit in MTU4.TMDR1 is set to 1 and MTU4.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.36 TIORH (MTU6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU6.TGRA Function	MTIOC6A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.37 TIORL (MTU6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU6.TGRC Function	MTIOC6C Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*[1]	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the BFA bit in MTU6.TMDR1 is set to 1 and MTU6.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.38 TIORH (MTU7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU7.TGRA Function	MTIOC7A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Table 16.39 TIORL (MTU7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU7.TGRC Function	MTIOC7C Pin Function
0	0	0	0	Output compare register*1	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register*1	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don't care

Note 1. When the BFA bit in MTU7.TMDR1 is set to 1 and MTU7.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.40 TIORH (MTU8)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MTU8.TGRA Function	MTIOC8A Pin Function
0	0	0	0	Output compare register	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	0	0	0	Input capture register	Input capture at rising edge.
1	0	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Table 16.41 TIORL (MTU8)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MTU8.TGRC Function	MTIOC8C Pin Function
0	0	0	0	Output compare register*[1]	Output prohibited
0	0	0	1		Initial output is low. Low output at compare match.
0	0	1	0		Initial output is low. High output at compare match.
0	0	1	1		Initial output is low. Toggle output at compare match.
0	1	0	0		Output prohibited
0	1	0	1		Initial output is high. Low output at compare match.
0	1	1	0		Initial output is high. High output at compare match.
0	1	1	1		Initial output is high. Toggle output at compare match.
1	x	0	0	Input capture register	Input capture at rising edge.
1	x	0	1		Input capture at falling edge.
1	x	1	x		Input capture at both edges.

Remarks: x: Don’t care

Note 1. When the MTU8.TMDR1.BFA bit is set to 1 and the MTU8.TGRC register is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.42 TIORU, TIORV, and TIORW (MTU5)

Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC4	IOC3	IOC2	IOC1	IOC0	MTU5.TGRU, MTU5.TGRV, MTU5.TGRW Function	MTIC5U, MTIC5V, MTIC5W Pin Function
0	0	0	0	0	Output compare register	No function
0	0	0	0	1		Setting prohibited
0	0	0	1	x		Setting prohibited
0	0	1	x	x		Setting prohibited
0	1	x	x	x		Setting prohibited
1	0	0	0	0	Input capture register*[1]	Setting prohibited
1	0	0	0	1		Input capture at rising edge.
1	0	0	1	0		Input capture at falling edge.
1	0	0	1	1		Input capture at both edges.
1	0	1	x	x		Input capture on generation of compare match with MTU8.TGRC
1	1	0	0	0		Setting prohibited
1	1	0	0	1		Measurement of low pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	0	1	0		Measurement of low pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	0	1	1		Measurement of low pulse width of external input signal. Capture at crest and trough of complementary PWM mode.
1	1	1	0	0		Setting prohibited
1	1	1	0	1		Measurement of high pulse width of external input signal. Capture at trough in complementary PWM mode.
1	1	1	1	0		Measurement of high pulse width of external input signal. Capture at crest of complementary PWM mode.
1	1	1	1	1		Measurement of high pulse width of external input signal. Capture at crest and trough of complementary PWM mode.

Remarks: x: Don’t care

Note 1. Setting the IOC[4:0] bits to “H’19”, “H’1A”, “H’1B”, “H’1D”, “H’1E”, or “H’1F” is only allowed when the external pulse width measurement function is used or the dead time compensation function, in coordination with MTU6 and MTU7, is used. For details, refer to **Section 16.3.11, External Pulse Width Measurement**, and **Section 16.3.12, Dead Time Compensation**.

16.2.7 Timer Compare Match Clear Register (TCNTCMPCLR)

Address(es): MTU5.TCNTCMPCLR H'0_1000_1CB6

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CMPCLR5U	CMPCLR5V	CMPCLR5W
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CMPCLR5W	0	R/W	TCNT Compare Clear 5W 0: Disables MTU5.TCNTW to be cleared to H'0000 at MTU5.TCNTW and MTU5.TGRW compare match or input capture 1: Enables MTU5.TCNTW to be cleared to H'0000 at MTU5.TCNTW and MTU5.TGRW compare match or input capture
b1	CMPCLR5V	0	R/W	TCNT Compare Clear 5V 0: Disables MTU5.TCNTV to be cleared to H'0000 at MTU5.TCNTV and MTU5.TGRV compare match or input capture 1: Enables MTU5.TCNTV to be cleared to H'0000 at MTU5.TCNTV and MTU5.TGRV compare match or input capture
b2	CMPCLR5U	0	R/W	TCNT Compare Clear 5U 0: Disables MTU5.TCNTU to be cleared to H'0000 at MTU5.TCNTU and MTU5.TGRU compare match or input capture 1: Enables MTU5.TCNTU to be cleared to H'0000 at MTU5.TCNTU and MTU5.TGRU compare match or input capture
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TCNTCMPCLR specifies requests to clear MTU5.TCNTU, MTU5.TCNTV, and MTU5.TCNTW. The MTU has one TCNTCMPCLR (on MTU5).

16.2.8 Timer Interrupt Enable Register (TIER)

(1) MTU1.TIER, MTU2.TIER

Address(es): MTU1.TIER H'0_1000_1384, MTU2.TIER H'0_1000_1404

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU0.TIER, MTU3.TIER, MTU6.TIER

Address(es): MTU0.TIER H'0_1000_1304, MTU3.TIER H'0_1000_1208, MTU6.TIER H'0_1000_1A08

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(3) MTU4.TIER, MTU7.TIER

Address(es): MTU4.TIER H'0_1000_1209, MTU7.TIER H'0_1000_1A09

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(4) MTU8.TIER

Address(es): MTU8.TIER H'0_1000_1604

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TGIEA	0	R/W	TGR Interrupt Enable A 0: Interrupt requests (TGIA) disabled 1: Interrupt requests (TGIA) enabled
b1	TGIEB	0	R/W	TGR Interrupt Enable B 0: Interrupt requests (TGIB) disabled 1: Interrupt requests (TGIB) enabled
b2	TGIEC	0	R/W	TGR Interrupt Enable C 0: Interrupt requests (TGIC) disabled 1: Interrupt requests (TGIC) enabled
b3	TGIED	0	R/W	TGR Interrupt Enable D 0: Interrupt requests (TGID) disabled 1: Interrupt requests (TGID) enabled
b4	TCIEV	0	R/W	Overflow Interrupt Enable 0: Interrupt requests (TCIV) disabled 1: Interrupt requests (TCIV) enabled
b5	TCIEU	0	R/W	Underflow Interrupt Enable 0: Interrupt requests (TCIU) disabled 1: Interrupt requests (TCIU) enabled
b6	TTGE2	0	R/W	A/D Converter Start Request Enable 2 0: A/D converter start request generation by MTUn.TCNT underflow (trough) disabled 1: A/D converter start request generation by MTUn.TCNT underflow (trough) enabled
b7	TTGE	0	R/W	A/D Converter Start Request Enable 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled

Note: n = 4, 7

The TIER register enables or disables interrupt requests from each channel. The MTU has a total of ten TIER registers, two for MTU0 and one each for MTU1 to MTU8.

TGIEA and TGIEB Bits (TGR Interrupt Enable A and B)

Each bit enables or disables interrupt requests (TGIn) (n = A, B).

TGIEC and TGIED Bits (TGR Interrupt Enable C and D)

Each bit enables or disables an interrupt request (TGIn) (n = C, D).

In MTU1 and MTU2, these bits are reserved. They are read as 0. The write value should be 0.

TCIEV Bit (Overflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIV).

TCIEU Bit (Underflow Interrupt Enable)

This bit enables or disables interrupt requests (TCIU).

In MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

TTGE2 Bit (A/D Converter Start Request Enable 2)

This bit enables or disables generation of A/D converter start requests by MTUn.TCNT underflow (trough) in complementary PWM mode (n = 4, 7).

In MTU0 to MTU3, MTU6, and MTU8, this bit is reserved. It is read as 0. The write value should be 0.

TTGE Bit (A/D Converter Start Request Enable)

This bit enables or disables generation of A/D converter start requests by TGRA input capture/compare match. MTU8 is a reserved bit. It is read as 0. The write value should be 0.

(5) MTU0.TIER2

Address(es): MTU0.TIER2 H'0_1000_1324

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE2	—	—	—	—	—	TGIEF	TGIEE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TGIEE	0	R/W	TGR Interrupt Enable E 0: Interrupt requests (TGIE) disabled 1: Interrupt requests (TGIE) enabled
b1	TGIEF	0	R/W	TGR Interrupt Enable F 0: Interrupt requests (TGIF) disabled 1: Interrupt requests (TGIF) enabled
b6 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b7	TTGE2	0	R/W	A/D Converter Start Request Enable 2 0: A/D converter start request generation by compare match between MTU0.TCNT and MTU0.TGRE disabled 1: A/D converter start request generation by compare match between MTU0.TCNT and MTU0.TGRE enabled

TGIEE and TGIEF Bits (TGR Interrupt Enable E and F)

Each bit enables or disables interrupt requests by compare match between MTU0.TCNT and MTU0.TGRn (n = E, F).

TTGE2 Bit (A/D Converter Start Request Enable 2)

This bit enables or disables AD converter start requests by compare match between TCNT and TGRE in MTU0.

(6) MTU5.TIER

Address(es): MTU5.TIER H'0_1000_1CB2

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TGIE5U	TGIE5V	TGIE5W
Initial Value	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TGIE5W	0	R/W	TGR Interrupt Enable 5W 0: Interrupt requests TGIW5 disabled 1: Interrupt requests TGIW5 enabled
b1	TGIE5V	0	R/W	TGR Interrupt Enable 5V 0: Interrupt requests TGIV5 disabled 1: Interrupt requests TGIV5 enabled
b2	TGIE5U	0	R/W	TGR Interrupt Enable 5U 0: Interrupt requests TGIU5 disabled 1: Interrupt requests TGIU5 enabled
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TGIE5n Bits (TGR Interrupt Enable 5n)

Each bit enables or disables interrupt requests (TGIn5) (n = U, V, W).

16.2.9 Timer Status Register (TSR)

(1) MTU1.TSR, MTU2.TSR

Address(es): MTU1.TSR H'0_1000_1385, MTU2.TSR H'0_1000_1405

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Initial Value	1	1	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU3.TSR, MTU4.TSR, MTU6.TSR, MTU7.TSR

Address(es): MTU3.TSR H'0_1000_122C, MTU4.TSR H'0_1000_122D, MTU6.TSR H'0_1000_1A2C,
MTU7.TSR H'0_1000_1A2D

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	—	—	—	—	—
Initial Value	1	1	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b5 to b0	—	All 0	R/W	Reserved The read value is undefined. The write value should be 1.
b6	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b7	TCFD	1	R	Count Direction Flag 0: TCNT counts down 1: TCNT counts up

TSR indicates the states of each of the channels. The MTU has a total of six TSR registers, one each for MTU1 to MTU4, MTU6, and MTU7.

TCFD Flag (Count Direction Flag)

Status flag that indicates the direction in which TCNT is counting in MTU1 to MTU4, MTU6, and MTU7.

16.2.10 Timer Buffer Operation Transfer Mode Register (TBTM)

(1) MTU0.TBTM

Address(es): MTU0.TBTM H'0_1000_1326

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TTSE	TTSB	TTSA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) MTU3.TBTM, MTU4.TBTM, MTU6.TBTM, MTU7.TBTM

Address(es): MTU3.TBTM H'0_1000_1238, MTU4.TBTM H'0_1000_1239, MTU6.TBTM H'0_1000_1A38,
MTU7.TBTM H'0_1000_1A39

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TTSB	TTSA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TTSA	0	R/W	Timing Select A 0: When compare match A occurs in each channel, data is transferred from TGRC to TGRA 1: When TCNT is cleared in each channel, data is transferred from TGRC to TGRA
b1	TTSB	0	R/W	Timing Select B 0: When compare match B occurs in each channel, data is transferred from TGRD to TGRB 1: When TCNT is cleared in each channel, data is transferred from TGRD to TGRB
b2	TTSE	0	R/W	Timing Select E 0: When compare match E occurs in MTU0, data is transferred from MTU0.TGRF to MTU0.TGRE 1: When MTU0.TCNT is cleared in MTU0, data is transferred from MTU0.TGRF to MTU0.TGRE
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TBTM specifies the timing for transferring data from the buffer register to the timer general register in PWM mode. The MTU has a total of five TBTM registers, one each for MTU0, MTU3, MTU4, MTU6, and MTU7.

TTSA Bit (Timing Select A)

This bit specifies the timing for transferring data from TGRC to TGRA in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSA bit in the channel to 1.

TTSB Bit (Timing Select B)

This bit specifies the timing for transferring data from TGRD to TGRB in each channel when they are used together for buffer operation. When a channel is not set to PWM mode, do not set the TTSB bit in the channel to 1.

TTSE Bit (Timing Select E)

This bit specifies the timing for transferring data from MTU0.TGRF to MTU0.TGRE when they are used together for buffer operation.

In MTU3, MTU4, MTU6, and MTU7, this bit is reserved. It is read as 0 and the write value should be 0. When a channel is not set to PWM mode, do not set the TTSE bit in the channel to 1.

16.2.11 Timer Input Capture Control Register (TICCR)

Address(es): MTU1.TICCR H'0_1000_1390

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	I2BE	I2AE	I1BE	I1AE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	I1AE	0	R/W	Input Capture Enable 0: Does not include the MTIOC1A pin in the MTU2.TGRA input capture conditions 1: Includes the MTIOC1A pin in the MTU2.TGRA input capture conditions
b1	I1BE	0	R/W	Input Capture Enable 0: Does not include the TMTIOC1B pin in the MTU2.TGRB input capture conditions 1: Includes the MTIOC1B pin in the MTU2.TGRB input capture conditions
b2	I2AE	0	R/W	Input Capture Enable 0: Does not include the MTIOC2A pin in the MTU1.TGRA input capture conditions 1: Includes the MTIOC2A pin in the MTU1.TGRA input capture conditions
b3	I2BE	0	R/W	Input Capture Enable 0: Does not include the MTIOC2B pin in the MTU1.TGRB input capture conditions 1: Includes the MTIOC2B pin in the MTU1.TGRB input capture conditions
b7 to b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TICCR specifies input capture conditions when MTU1.TCNT and MTU2.TCNT are cascaded. The MTU has one TICCR for MTU1.

16.2.12 Timer Synchronous Clear Register (TSYCR)

Address(es): MTU6.TSYCR H'0_1000_1A50

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CE0A	CE0B	CE0C	CE0D	CE1A	CE1B	CE2A	CE2B
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CE2B	0	R/W	Clear Enable 2B 0: Disables counter clearing by the MTU2.TGIB2 interrupt generation timing. 1: Enables counter clearing by the MTU2.TGIB2 interrupt generation timing.
b1	CE2A	0	R/W	Clear Enable 2A 0: Disables counter clearing by the MTU2.TGIA2 interrupt generation timing*[1]. 1: Enables counter clearing by the MTU2.TGIA2 interrupt generation timing*[1].
b2	CE1B	0	R/W	Clear Enable 1B 0: Disables counter clearing by the MTU1.TGIB1 interrupt generation timing*[1]. 1: Enables counter clearing by the MTU1.TGIB1 interrupt generation timing*[1].
b3	CE1A	0	R/W	Clear Enable 1A 0: Disables counter clearing by the MTU1.TGIA1 interrupt generation timing*[1]. 1: Enables counter clearing by the MTU1.TGIA1 interrupt generation timing*[1].
b4	CE0D	0	R/W	Clear Enable 0D 0: Disables counter clearing by the MTU0.TGID0 interrupt generation timing*[1]. 1: Enables counter clearing by the MTU0.TGID0 interrupt generation timing*[1].
b5	CE0C	0	R/W	Clear Enable 0C 0: Disables counter clearing by the MTU0.TGIC0 interrupt generation timing*[1]. 1: Enables counter clearing by the MTU0.TGIC0 interrupt generation timing*[1].
b6	CE0B	0	R/W	Clear Enable 0B 0: Disables counter clearing by the MTU0.TGIB0 interrupt generation timing*[1]. 1: Enables counter clearing by the MTU0.TGIB0 interrupt generation timing*[1].
b7	CE0A	0	R/W	Clear Enable 0A 0: Disables counter clearing by the MTU0.TGIA0 interrupt generation timing*[1]. 1: Enables counter clearing by the MTU0.TGIA0 interrupt generation timing*[1].

Note 1. This does not depend on the TIERn.TGIEm bit setting. (n = 0, 1, 2; m = A, B, C, D)

TSYCR specifies synchronous clear conditions for MTU6.TCNT and MTU7.TCNT. The MTU has one TSYCR for MTU1.

CEnm Bits (Clear Enable nm; n = 0, 1, 2; m = A, B, C, D)

These bits enable or disable counter clearing by the MTUn.TGImn interrupt generation timing.

16.2.13 Timer Counter (TCNT)

(1) MTU0.TCNT to MTU7.TCNT

Address(es): MTU0.TCNT H'0_1000_1306, MTU1.TCNT H'0_1000_1386, MTU2.TCNT H'0_1000_1406, MTU3.TCNT H'0_1000_1210, MTU4.TCNT H'0_1000_1212, MTU5.TCNTU H'0_1000_1C80, MTU5.TCNTV H'0_1000_1C90, MTU5.TCNTW H'0_1000_1CA0, MTU6.TCNT H'0_1000_1A10, MTU7.TCNT H'0_1000_1A12

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCNT must not be accessed in eight bits; it should be accessed in 16 bits.

(2) MTU8.TCNT

Address(es): MTU8.TCNT H'0_1000_1608

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCNT must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

MTU0.TCNT to MTU7.TCNT are 16-bit readable/writable counters and MTU8.TCNT is a 32-bit readable/writable counter. The MTU has a total of 11 TCNT counters, one each for MTU0 to MTU4, MTU6, MTU7, and MTU8 and three (TCNTU, TCNTV, and TCNTW) for MTU5. The TCNT counters in MTU0 to MTU4, MTU6, and MTU7 are initialized to H’0000 by a reset, and the TCNT counter in MTU8 is initialized to H’00000000 by a reset. The TCNTU, TCNTV, and TCNTW counters in MTU5 are initialized to H’0000 by a reset. In MTU0 to MTU4, MTU6, and MTU7, the TCNT counters must not be accessed in 8-bit units; they should be accessed in 16-bit units. The MTU8.TCNT counter must not be accessed in 8- or 16-bit units; it should be accessed in 32-bit units.

The MTU1.TCNT and MTU2.TCNT counters are read as H’0000 when TMDR3.LWA is 1. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details.

16.2.14 Timer Longword Counter (TCNTLW)

Address(es): MTU1.TCNTLW H’0_1000_13A0

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCNTLW must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The TCNTLW counter is a 32-bit readable/writable counter. Only one counter of this type is provided, and is formed by combining MTU1.TCNT and MTU2.TCNT. Such operation is only effective when TMDR3.LWA is 1. The TCNTLW counter is initialized to H’0000 0000 by a reset. This counter is read as H’0000 0000 when TMDR3.LWA is 0. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details. This register can only be used in 32-bit phase counting mode.

16.2.15 Timer General Register (TGR)

(1) MTU0.TGR to MTU7.TGR

Address(es): MTU0.TGRA H'0_1000_1308, MTU0.TGRB H'0_1000_130A, MTU0.TGRC H'0_1000_130C,
MTU0.TGRD H'0_1000_130E, MTU0.TGRE H'0_1000_1320, MTU0.TGRF H'0_1000_1322,
MTU1.TGRA H'0_1000_1388, MTU1.TGRB H'0_1000_138A, MTU2.TGRA H'0_1000_1408,
MTU2.TGRB H'0_1000_140A, MTU3.TGRA H'0_1000_1218, MTU3.TGRB H'0_1000_121A,
MTU3.TGRC H'0_1000_1224, MTU3.TGRD H'0_1000_1226, MTU3.TGRE H'0_1000_1272,
MTU4.TGRA H'0_1000_121C, MTU4.TGRB H'0_1000_121E, MTU4.TGRC H'0_1000_1228,
MTU4.TGRD H'0_1000_122A, MTU4.TGRE H'0_1000_1274, MTU4.TGRF H'0_1000_1276,
MTU5.TGRU H'0_1000_1C82, MTU5.TGRV H'0_1000_1C92, MTU5.TGRW H'0_1000_1CA2,
MTU6.TGRA H'0_1000_1A18, MTU6.TGRB H'0_1000_1A1A, MTU6.TGRC H'0_1000_1A24,
MTU6.TGRD H'0_1000_1A26, MTU6.TGRE H'0_1000_1A72, MTU7.TGRA H'0_1000_1A1C,
MTU7.TGRB H'0_1000_1A1E, MTU7.TGRC H'0_1000_1A28, MTU7.TGRD H'0_1000_1A2A,
MTU7.TGRE H'0_1000_1A74, MTU7.TGRF H'0_1000_1A76

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TGR must not be accessed in eight bits; it should be accessed in 16 bits. The initial value of TGR is H'FFFF.

(2) MTU8.TGR

Address(es): MTU8.TGRA H'0_1000_160C, MTU8.TGRB H'0_1000_1610, MTU8.TGRC H'0_1000_1614, MTU8.TGRD H'0_1000_1618

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TGR must not be accessed in 8 or 16 bits; it should be accessed in 32 bits.

The MTU0.TGR to MTU7.TGR registers are 16-bit readable/writable registers; the MTU8.TGR register is a 32-bit readable/writable register. The MTU has a total of 39 TGR registers, six for MTU0, two each for MTU1 and MTU2, five each for MTU3 and MTU6, six each for MTU4 and MTU7, three for MTU5 and four for MTU8.

The TGRA, TGRB, TGRC, and TGRD registers function as either output compare or input capture registers. The TGRC and TGRD registers for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8 can also be designated for operation as buffer registers. TGR buffer register combinations are TGRA and TGRC, and TGRB and TGRD.

MTU0.TGRE and MTU0.TGRF function as compare registers. When the MTU0.TCNT count matches the MTU0.TGRE value, an A/D converter start request can be issued. The TGRF register can also be designated for operation as a buffer register. TGR buffer register combination is TGRE and TGRF. MTU5.TGRU, MTU5.TGRV, and MTU5.TGRW function as compare match, input capture, or external pulse width measurement registers.

The MTU1.TGRA, MTU2.TGRA, MTU1.TGRB, and MTU2.TGRB registers are read as H'0000 when TMDR3.LWA is 1. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details.

16.2.16 Timer Longword General Registers (TGRALW and TGRBLW)

Address(es): MTU1.TGRALW H'0_1000_13A4, MTU1.TGRBLW H'0_1000_13A8

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TGRALW and TGRBLW must not be accessed in 8 or 16 bits; they should be accessed in 32 bits.

The TGRALW (TGRBLW) register is a 32-bit readable/writable register. Two general registers of this type are provided, and are formed by combining the TGRA (TGRB) registers in MTU1 and MTU2. Such operation is only effective when TMDR3.LWA is 1.

The TGRALW (TGRBLW) register is initialized to H'FFFF FFFF by a reset, but it is read as H'0000 0000 when TMDR3.LWA is 0. See **Section 16.2.5, Timer Mode Register 3 (TMDR3)** for details.

The TGRALW (TGRBLW) register functions as an output compare or input capture register when TMDR3.LWA is 1. This register can only be used in 32-bit phase counting mode.

16.2.17 Timer Start Registers (TSTRA, TSTRB, and TSTR)

(1) MTU.TSTRA (MTU0, MTU1, MTU2, MTU3, MTU4, MTU8)

Address(es): MTU.TSTRA H'0_1000_1280

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CST4	CST3	—	—	CST8	CST2	CST1	CST0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CST0	0	R/W	Counter Start 0 0: MTU0.TCNT counting is stopped 1: MTU0.TCNT performs count operation
b1	CST1	0	R/W	Counter Start 1 0: MTU1.TCNT counting is stopped 1: MTU1.TCNT performs count operation
b2	CST2	0	R/W	Counter Start 2 0: MTU2.TCNT counting is stopped 1: MTU2.TCNT performs count operation
b3	CST8	0	R/W	Counter start 8 0: MTU8.TCNT counting is stopped 1: MTU8.TCNT performs count operation.
b5, b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	CST3	0	R/W	Counter Start 3 0: MTU3.TCNT counting is stopped 1: MTU3.TCNT performs count operation
b7	CST4	0	R/W	Counter Start 4 0: MTU4.TCNT counting is stopped 1: MTU4.TCNT performs count operation

Note: When 1 is written to a bit in TCSYSTR, the corresponding bit in TSTRA is also set to 1 automatically.

The TSTRA register starts or stops TCNT operation in MTU0 to MTU4 and MTU8. TSTRA starts or stops TCNT operation in MTU0 to MTU4.

TSTRB starts or stops TCNT operation in MTU6 and MTU7. TSTR starts or stops TCNT operation in MTU5.

Before setting the operating mode in TMDR1 or setting the TCNT count clock in TCR, be sure to stop the TCNT counting.

CSTn Bits (Counter Start n) (n = 0, 1, 2, 3, 4, 8)

Each bit starts or stops TCNT in the corresponding channel.

If 0 is written to the CSTn bit during operation with the MTIOC pin designated for output, the counter stops. In this case, the initial output value specified by the TOCR1A or TOCR2A register is output from the MTIOC pin in complementary PWM mode or reset-synchronized PWM mode.

In the other modes, the level of the output compare signal from the MTIOC pin is retained. If TIOR is written to while the CSTn bit is 0, the level of output from the pin will be updated to the specified initial output value.

(2) MTU.TSTRB(MTU6, MTU7)

Address(es): MTU.TSTRB H'0_1000_1A80

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CST7	CST6	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b5 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	CST6	0	R/W	Counter Start 6 0: MTU6.TCNT counting is stopped 1: MTU6.TCNT performs count operation
b7	CST7	0	R/W	Counter Start 7 0: MTU7.TCNT counting is stopped 1: MTU7.TCNT performs count operation

Note: When 1 is written to a bit in TCSYSTR, the corresponding bit in TSTRB is also set to 1 automatically.

CSTn Bits (Counter Start n) (n = 6, 7)

Each bit starts or stops TCNT in the corresponding channel.

If 0 is written to the CSTn bit during operation with the MTIOC pin designated for output, the counter stops. In this case, the initial output value specified by the TOCR1B or TOCR2B register is output from the MTIOC pin in complementary PWM mode or reset-synchronized PWM mode.

In the other modes, the level of the output compare signal from the MTIOC pin is retained. If TIOR is written to while the CSTn bit is 0, the level of output from the pin will be updated to the specified initial output value.

(3) MTU5.TSTR(MTU5)

Address(es): MTU5.TSTR H'0_1000_1CB4

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CSTU5	CSTV5	CSTW5
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CSTW5	0	R/W	Counter Start W5 0: MTU5.TCNTW counting is stopped 1: MTU5.TCNTW performs count operation
b1	CSTV5	0	R/W	Counter Start V5 0: MTU5.TCNTV counting is stopped 1: MTU5.TCNTV performs count operation
b2	CSTU5	0	R/W	Counter Start U5 0: MTU5.TCNTU counting is stopped 1: MTU5.TCNTU performs count operation
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

16.2.18 Timer Synchronous Registers (TSYRA and TSYRB)

(1) MTU.TSYRA(MTU0, MTU1, MTU2, MTU3, MTU4)

Address(es): MTU.TSYRA H'0_1000_1281

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	SYNC0	0	R/W	Timer Synchronous Operation 0 0: MTU0.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU0.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b1	SYNC1	0	R/W	Timer Synchronous Operation 1 0: MTU1.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU1.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b2	SYNC2	0	R/W	Timer Synchronous Operation 2 0: MTU2.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU2.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b5 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	SYNC3	0	R/W	Timer Synchronous Operation 3 0: MTU3.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU3.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b7	SYNC4	0	R/W	Timer Synchronous Operation 4 0: MTU4.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU4.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)

TSYRA selects independent operation or synchronous operation of TCNT in MTU0 to MTU4. TSYRB selects independent operation or synchronous operation of TCNT in MTU6 and MTU7. A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

SYNCn Bits (Timer Synchronous Operation n) (n = 0, 1, 2, 3, 4)

Each bit selects whether operation is independent of or synchronized with other channels.

When synchronous operation is selected, the TCNT synchronous setting of multiple channels and synchronous clearing by counter clearing on another channel are possible.

To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of the TCR.CCLR[2:0] bits.

(2) MTU.TSYRB(MTU6, MTU7)

Address(es): MTU.TSYRB H'0_1000_1A81

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC7	SYNC6	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b5 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	SYNC6	0	R/W	Timer Synchronous Operation 6 0: MTU6.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU6.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)
b7	SYNC7	0	R/W	Timer Synchronous Operation 7 0: MTU7.TCNT operates independently (TCNT setting/clearing is not related to other channels). 1: MTU7.TCNT performs synchronous operation. (TCNT synchronous setting/synchronous clearing is enabled.)

SYNCn Bits (Timer Synchronous Operation n) (n = 6, 7)

Each bit selects whether operation is independent of or synchronized with other channels.

When synchronous operation is selected, the TCNT synchronous setting of multiple channels and synchronous clearing by counter clearing on another channel are possible.

To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits TCR.CCLR[2:0].

16.2.19 Timer Counter Synchronous Start Register (TCSYSTR)

Address(es): MTU.TCSYSTR H'0_1000_1282

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	SCH0	SCH1	SCH2	SCH3	SCH4	—	SCH6	SCH7
Initial Value	0	0	0	0	0	0	0	0
R/W	R/(W)[*1]	R/(W)[*1]	R/(W)[*1]	R/(W)[*1]	R/(W)[*1]	R	R/(W)[*1]	R/(W)[*1]

Bit	Bit Name	Initial Value	R/W	Description
b0	SCH7	0	R/(W)[*1]	Synchronous Start 7 0: Does not specify synchronous start for MTU7.TCNT 1: Specifies synchronous start for MTU7.TCNT
b1	SCH6	0	R/(W)[*1]	Synchronous Start 6 0: Does not specify synchronous start for MTU6.TCNT 1: Specifies synchronous start for MTU6.TCNT
b2	—	0	R	Reserved This bit is read as 0. The write value should be 0.
b3	SCH4	0	R/(W)[*1]	Synchronous Start 4 0: Does not specify synchronous start for MTU4.TCNT 1: Specifies synchronous start for MTU4.TCNT
b4	SCH3	0	R/(W)[*1]	Synchronous Start 3 0: Does not specify synchronous start for MTU3.TCNT 1: Specifies synchronous start for MTU3.TCNT
b5	SCH2	0	R/(W)[*1]	Synchronous Start 2 0: Does not specify synchronous start for MTU2.TCNT 1: Specifies synchronous start for MTU2.TCNT
b6	SCH1	0	R/(W)[*1]	Synchronous Start 1 0: Does not specify synchronous start for MTU1.TCNT 1: Specifies synchronous start for MTU1.TCNT
b7	SCH0	0	R/(W)[*1]	Synchronous Start 0 0: Does not specify synchronous start for MTU0.TCNT 1: Specifies synchronous start for MTU0.TCNT

Note 1. Only 1 can be written to this bit. It is automatically cleared to 0 when counting begins.

TCSYSTR specifies synchronous start of the counters.

SCH7 Bit (Synchronous Start 7)

This bit controls synchronous start of TCNT in MTU7. [Clearing condition]

- When the CST7 bit in TSTRB is set to 1 while SCH7 = 1

SCH6 Bit (Synchronous Start 6)

This bit controls synchronous start of TCNT in MTU6. [Clearing condition]

- When the CST6 bit in TSTRB is set to 1 while SCH6 = 1

SCH4 Bit (Synchronous Start 4)

This bit controls synchronous start of TCNT in MTU4. [Clearing condition]

- When the CST4 bit in TSTRA is set to 1 while SCH4 = 1

SCH3 Bit (Synchronous Start 3)

This bit controls synchronous start of TCNT in MTU3. [Clearing condition]

- When the CST3 bit in TSTRA is set to 1 while SCH3 = 1

SCH2 Bit (Synchronous Start 2)

This bit controls synchronous start of TCNT in MTU2. [Clearing condition]

- When the CST2 bit in TSTRA is set to 1 while SCH2 = 1

SCH1 Bit (Synchronous Start 1)

This bit controls synchronous start of TCNT in MTU1. [Clearing condition]

- When the CST1 bit in TSTRA is set to 1 while SCH1 = 1

SCH0 Bit (Synchronous Start 0)

This bit controls synchronous start of TCNT in MTU0. [Clearing condition]

- When the CST0 bit in TSTRA is set to 1 while SCH0 = 1

16.2.20 Timer Read/Write Enable Registers (TRWERA and TRWERB)

Address(es): MTU.TRWERA H'0_1000_1284, MTU.TRWERB H'0_1000_1A84

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	RWE
Initial Value	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	RWE	1	R/W	Read/Write Enable 0: Read/write access to the registers is disabled 1: Read/write access to the registers is enabled
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TRWERA enables or disables access to the registers and counters that have write-protection capability against accidental modification in MTU3 and MTU4.

TRWERB enables or disables access to the registers and counters that have write-protection capability against accidental modification in MTU6 and MTU7.

RWE Bit (Read/Write Enable)

This bit enables or disables access to the registers that have write-protection capability against accidental modification.

[Clearing condition]

When 0 is written to the RWE bit after reading RWE = 1

- Registers and Counters having Write-Protection Capability against Accidental Modification (TRWERA)

  24 registers: MTUn.TCR, MTUn.TCR2, MTUn.TMDR1, MTUn.TIORH, MTUn.TIORL, MTUn.TIER, MTUn.TGRA, MTUn.TGRB, TOERA, TOCR1A, TOCR2A, TGCRA, TCDRA, TDDRA, and MTUn.TCNT (n = 3, 4)

- Registers and Counters having Write-Protection Capability against Accidental Modification (TRWERB)

  23 registers: MTUn.TCR, MTUn.TCR2, MTUn.TMDR1, MTUn.TIORH, MTUn.TIORL, MTUn.TIER, MTUn.TGRA, MTUn.TGRB, TOERB, TOCR1B, TOCR2B, TCDRB, TDDRB, and MTUn.TCNT (n = 6, 7)

16.2.21 Timer Output Master Enable Registers (TOERA and TOERB)

(1) MTU.TOERA

Address(es): MTU.TOERA H'0_1000_120A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
Initial Value	1	1	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OE3B	0	R/W	Master Enable MTIOC3B 0: MTU output is disabled[*1] 1: MTU output is enabled
b1	OE4A	0	R/W	Master Enable MTIOC4A 0: MTU output is disabled[*1] 1: MTU output is enabled
b2	OE4B	0	R/W	Master Enable MTIOC4B 0: MTU output is disabled[*1] 1: MTU output is enabled
b3	OE3D	0	R/W	Master Enable MTIOC3D 0: MTU output is disabled[*1] 1: MTU output is enabled
b4	OE4C	0	R/W	Master Enable MTIOC4C 0: MTU output is disabled[*1] 1: MTU output is enabled
b5	OE4D	0	R/W	Master Enable MTIOC4D 0: MTU output is disabled[*1] 1: MTU output is enabled
b7, b6	—	All 1	R/W	Reserved These bits are read as 1. The write value should be 1.

Note 1. To output the inactive level from each pin when the MTU output is set to disabled, first set the port mode register (PMn) and port register (Pn) of I/O ports to output the inactive level from general I/O ports, and then set the port mode control register (PMCn) to use general I/O ports. For details, refer to **Section 41**, **General Purpose Input Output Port**.

TOERA enables or disables output settings for output pins MTIOC4D, MTIOC4C, MTIOC3D, MTIOC4B, MTIOC4A, and MTIOC3B.

These pins do not output values correctly if the bits of the TOERA register are not appropriately set. Write a desired value to the TOERA register before setting up the TIOR registers in MTU3 and MTU4.

Set the TOERA register after clearing the CST3 and CST4 bits in the TSTRA register to 0 (see **Figure 16.44** and **Figure 16.48**).

(2) MTU.TOERB

Address(es): MTU.TOERB H'0_1000_1A0A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE7D	OE7C	OE6D	OE7B	OE7A	OE6B
Initial Value	1	1	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OE6B	0	R/W	Master Enable MTIOC6B 0: MTU output is disabled*[1] 1: MTU output is enabled
b1	OE7A	0	R/W	Master Enable MTIOC7A 0: MTU output is disabled*[1] 1: MTU output is enabled
b2	OE7B	0	R/W	Master Enable MTIOC7B 0: MTU output is disabled*[1] 1: MTU output is enabled
b3	OE6D	0	R/W	Master Enable MTIOC6D 0: MTU output is disabled*[1] 1: MTU output is enabled
b4	OE7C	0	R/W	Master Enable MTIOC7C 0: MTU output is disabled*[1] 1: MTU output is enabled
b5	OE7D	0	R/W	Master Enable MTIOC7D 0: MTU output is disabled*[1] 1: MTU output is enabled
b7, b6	—	All 1	R/W	Reserved These bits are read as 1. The write value should be 1.

Note 1. To output the inactive level from each pin when the MTU output is set to disabled, first set the port mode register (PMn) and port register (Pn) of I/O ports to output the inactive level from general I/O ports, and then set the port mode control register (PMCn) to use general I/O ports. For details, refer to **Section 41**, **General Purpose Input Output Port**.

TOERB enables or disables output settings for output pins MTIOC7D, MTIOC7C, MTIOC6D, MTIOC7B, MTIOC7A, and MTIOC6B.

These pins do not output values correctly if the bits of the TOERB register are not appropriately set. Write a desired value to the TOERB register before setting up the TIOR registers in MTU6 and MTU7.

Set the TOERB register after clearing the CST6 and CST7 bits in the TSTRB register to 0 (see **Figure 16.44** and **Figure 16.48**).

16.2.22 Timer Output Control Registers 1 (TOCR1A and TOCR1B)

Address(es): MTU.TOCR1A H'0_1000_120E, MTU.TOCR1B H'0_1000_1A0E

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OLSP	0	R/W	Output Level Select P*1 *3 See **Table 16.43**.
b1	OLSN	0	R/W	Output Level Select N*1 *3 See **Table 16.44**.
b2	TOCS	0	R/W	TOC Select 0: TOCR1j setting is selected (j = A, B) 1: TOCR2j setting is selected
b3	TOCL	0	R/W	TOC Register Write Protection*2 *4 0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled
b5, b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b6	PSYE	0	R/W	PWM Synchronous Output Enable 0: Toggle output is disabled 1: Toggle output is enabled
b7	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Note 1. Clearing the TOCR1j.TOCS bit to 0 makes this bit setting valid.

Note 2. Setting the TOCR1j.TOCL bit to 1 prevents accidental modification when the CPU goes out of control.

Note 3. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSP bit is valid.

Note 4. This bit can be set to 1 only once after a reset. After 1 is written, 0 cannot be written to the bit.

TOCR1A and TOCR1B enable or disable PWM-synchronized toggle output in complementary PWM mode and reset-synchronized PWM mode, and control inversion of PWM output level.

OLSP Bit (Output Level Select P)

This bit selects the positive-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

OLSN Bit (Output Level Select N)

This bit selects the negative-phase output level in reset-synchronized PWM mode and complementary PWM mode. The initial output is selected while the counter is stopped.

TOCS Bit (TOC Select)

This bit selects either the TOCR1j or TOCR2j (j = A, B) setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode.

TOCL Bit (TOC Register Write Protection)

This bit enables or disables write access to the TOCS, OLSN, and OLSP bits in TOCR1j (j = A, B).

PSYE Bit (PWM Synchronous Output Enable)

This bit enables or disables toggle output synchronized with the PWM cycle.

Table 16.43 Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLSP	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 16.44 Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLSN	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Figure 16.3 shows an example of output in complementary PWM mode (one phase) when OLSN = 1 and OLSP = 1.

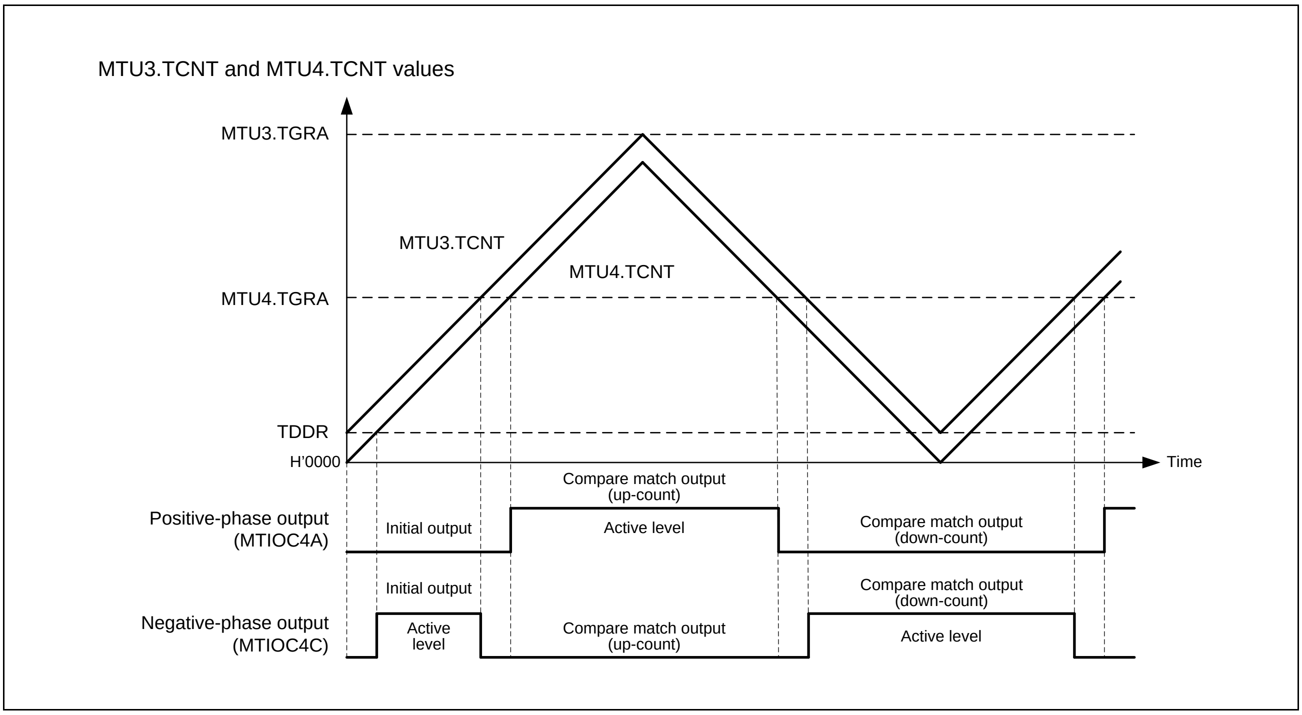


Figure 16.3 Example of Output in Complementary PWM Mode

16.2.23 Timer Output Control Registers 2 (TOCR2A and TOCR2B)

Address(es): MTU.TOCR2A H'0_1000_120F, MTU.TOCR2B H'0_1000_1A0F

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OLS1P	0	R/W	Output Level Select 1P[*1] [*2] This bit selects the output level on MTIOC3B or MTIOC6B in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.45**.
b1	OLS1N	0	R/W	Output Level Select 1N[*1] [*2] This bit selects the output level on MTIOC3D or MTIOC6D in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.46**.
b2	OLS2P	0	R/W	Output Level Select 2P[*1] [*2] This bit selects the output level on MTIOC4A or MTIOC7Ain reset-synchronized PWM mode and complementary PWM mode. See **Table 16.47**.
b3	OLS2N	0	R/W	Output Level Select 2N[*1] [*2] This bit selects the output level on MTIOC4C or MTIOC7C in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.48**.
b4	OLS3P	0	R/W	Output Level Select 3P[*1] [*2] This bit selects the output level on MTIOC4B or MTIOC7B in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.49**.
b5	OLS3N	0	R/W	Output Level Select 3N[*1] [*2] This bit selects the output level on MTIOC4D or MTIOC7D in reset-synchronized PWM mode and complementary PWM mode. See **Table 16.50**.
b7, b6	BF[1:0]	All 0	R/W	TOLBR Buffer Transfer Timing Select These bits select the timing for transferring data from TOLBRj to TOCR2j. See **Table 16.51** for details.

Note: j = A, B

Note 1. Setting the TOCR1j.TOCS bit to 1 makes this bit setting valid.

Note 2. If dead-time is not generated, the negative-phase output is the exact inverse of the positive-phase output. In this case, only the OLSiP bits are valid (i = 1 to 3).

TOCR2A and TOCR2B control inversion of PWM output level in complementary PWM mode and reset-synchronized PWM mode.

The initial output is selected while the counter is stopped.

Table 16.45 MTIOCmB Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLS1P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Note: m = 3, 6

Table 16.46 MTIOCmD Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLS1N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: m = 3, 6

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 16.47 MTIOCmA Output Level Select Function

Bit 2	Function			
			Compare Match Output	
OLS2P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Note: m = 4, 7

Table 16.48 MTIOCmC Output Level Select Function

Bit 3	Function			
			Compare Match Output	
OLS2N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: m = 4, 7

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 16.49 MTIOCmB Output Level Select Function

Bit 4	Function			
			Compare Match Output	
OLS3P	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Note: m = 4, 7

Table 16.50 MTIOCmD Output Level Select Function

Bit 5	Function			
			Compare Match Output	
OLS3N	Initial Output	Active Level	Up-Counting	Down-Counting
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: m = 4, 7

Note: The initial output value of negative-phase waveform changes to an active level after the dead time has passed since counting starts.

Table 16.51 Setting of TOCR2j.BF[1:0] Bits

Bit 7	Bit 6	Description	
BF1	BF0	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the buffer register (TOLBRj) to TOCR2j.	Does not transfer data from the buffer register (TOLBRj) to TOCR2j.
	1	Transfers data from the buffer register (TOLBRj) to TOCR2j at the crest of the MTUn.TCNT count.	Transfers data from the buffer register (TOLBRj) to TOCR2j when MTUm.TCNT or MTUn.TCNT is cleared.
1	0	Transfers data from the buffer register (TOLBRj) to TOCR2j at the trough of the MTUn.TCNT count.	Setting prohibited
	1	Transfers data from the buffer register (TOLBRj) to TOCR2j at the crest and trough of the MTUn.TCNT count.	Setting prohibited

Note: n = 4, 7;
m = 3, 6;
j = A, B

16.2.24 Timer Output Level Buffer Registers (TOLBRA and TOLBRB)

Address(es): MTU.TOLBRA H'0_1000_1236, MTU.TOLBRB H'0_1000_1A36

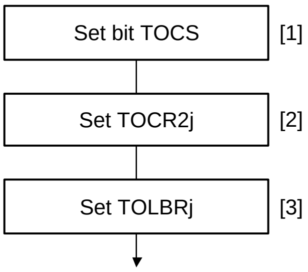
Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	OLS1P	0	R/W	Output Level Select 1P Specify the buffer value to be transferred to the OLS1P bit in TOCR2j.
b1	OLS1N	0	R/W	Output Level Select 1N Specify the buffer value to be transferred to the OLS1N bit in TOCR2j.
b2	OLS2P	0	R/W	Output Level Select 2P Specify the buffer value to be transferred to the OLS2P bit in TOCR2j.
b3	OLS2N	0	R/W	Output Level Select 2N Specify the buffer value to be transferred to the OLS2N bit in TOCR2j.
b4	OLS3P	0	R/W	Output Level Select 3P Specify the buffer value to be transferred to the OLS3P bit in TOCR2j.
b5	OLS3N	0	R/W	Output Level Select 3N Specify the buffer value to be transferred to the OLS3N bit in TOCR2j.
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note: j = A, B

TOLBRA and TOLBRB are buffer registers for TOCR2A and TOCR2B and specify the PWM output level in complementary PWM mode and reset-synchronized PWM mode.

Figure 16.4 shows an example of the PWM output level setting procedure in buffer operation.



[1] Set bit TOCS in TOCR1j (j = A or B) to 1 to enable the TOCR2j setting.

[2] Use bits TOCR2j.BF[1:0] to select the TOLBRj buffer transfer timing. Use bits OLS3N to OLS1N and OLS3P to OLS1P to specify the PWM output levels.

[3] The TOLBRj initial setting must be the same value as specified in bits OLS3N to OLS1N and OLS3P to OLS1P in TOCR2j.

Figure 16.4 Example of PWM Output Level Setting Procedure in Buffer Operation

16.2.25 Timer Gate Control Register A (TGCRA)

Address(es): MTU.TGCRA H'0_1000_120D

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	BDC	N	P	FB	WF	VF	UF
Initial Value	1	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	UF	0	R/W	Output Phase Switch
b1	VF	0	R/W	These bits turn on or off the positive-phase/negative-phase output. The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. See **Table 16.52**.
b2	WF	0	R/W	
b3	FB	0	R/W	External Feedback Signal Enable 0: Output is switched by external input (input sources are TGRA, TGRB, and TGRC input capture signals in MTU0) 1: Output is switched by software (TGCRA's UF, VF, and WF settings)
b4	P	0	R/W	Positive-Phase Output (P) Control 0: Level output 1: Reset-synchronized PWM or complementary PWM output
b5	N	0	R/W	Negative-Phase Output (N) Control 0: Level output 1: Reset-synchronized PWM or complementary PWM output
b6	BDC	0	R/W	Brushless DC Motor 0: Ordinary output 1: Functions of this register are effective.
b7	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.

TGCRA controls the output waveform necessary for brushless DC motor control in reset-synchronized PWM mode and complementary PWM mode. TGCRA register settings are ineffective for anything other than complementary PWM mode and reset-synchronized PWM mode.

UF, VF, and WF Bits (Output Phase Switch)

The setting of these bits is valid only when the FB bit is set to 1. In this case, the setting of b0 to b2 is used instead of the external input. See **Table 16.52** for details.

FB Bit (External Feedback Signal Enable)

This bit selects whether the positive-/negative-phase output is switched automatically with the TGRA, TGRB, and TGRC input capture signals in MTU0 or by writing 0 or 1 to bits 2 to 0 in TGCRA.

When the FB bit in the TGCRA register is 0, the output signals from MTU3 and MTU4 are switched automatically with the TGRA, TGRB, and TGRC input capture signals in MTU0.

P Bit (Positive-Phase Output (P) Control)

This bit selects either the level output or the reset-synchronized PWM/complementary PWM output for the positive-phase output pins (MTIOC3B, MTIOC4A, and MTIOC4B pins).

N Bit (Negative-Phase Output (N) Control)

This bit selects the level output or the reset-synchronized PWM/complementary PWM output for the negative-phase output pins (MTIOC3D, MTIOC4C, and MTIOC4D pins).

BDC Bit (Brushless DC Motor)

This bit selects whether to make the functions of TGCRA effective or ineffective.

Table 16.52 Output Level Select Function

Bit 2	Bit 1	Bit 0	Function					
			MTIOC3B	MTIOC4A	MTIOC4B	MTIOC3D	MTIOC4C	MTIOC4D
WF	VF	UF	U Phase	V Phase	W Phase	U Phase	V Phase	W Phase
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
		1	ON	OFF	OFF	OFF	OFF	ON
	1	0	OFF	ON	OFF	ON	OFF	OFF
		1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
		1	ON	OFF	OFF	OFF	ON	OFF
	1	0	OFF	OFF	ON	ON	OFF	OFF
		1	OFF	OFF	OFF	OFF	OFF	OFF

16.2.26 Timer Subcounters (TCNTSA and TCNTSB)

Address(es): MTU.TCNTSA H'0_1000_1220, MTU.TCNTSB H'0_1000_1A20

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note: TCNTSA and TCNTSB must not be accessed in eight bits; it should be accessed in 16 bits.

TCNTSA and TCNTSB are 16-bit read-only counters that are used only in complementary PWM mode. The initial value of TCNTSA and TCNTSB after a reset is H'0000.

16.2.27 Timer Cycle Data Registers (TCDRA and TCDRB)

Address(es): MTU.TCDRA H'0_1000_1214, MTU.TCDRB H'0_1000_1A14

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCDRA and TCDRB must not be accessed in eight bits; it should be accessed in 16 bits.

TCDRA and TCDRB are 16-bit readable/writable registers used only in complementary PWM mode. Set half the PWM carrier cycle as the TCDRA and TCDRB values. The TCDRA and TCDRB registers are constantly compared with the TCNTSA and TCNTSB counters in complementary PWM mode, respectively. When a match occurs, the TCNTSA and TCNTSB counters switch the count direction (down-count to up-count). The initial value of TCDRA and TCDRB after a reset is H'FFFF.

16.2.28 Timer Cycle Buffer Registers (TCBRA and TCBRB)

Address(es): MTU.TCBRA H’0_1000_1222, MTU.TCBRB H’0_1000_1A22

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TCBRA and TCBRB must not be accessed in eight bits; it should be accessed in 16 bits.

TCBRA and TCBRB are 16-bit readable/writable registers, used only in complementary PWM mode, that function as buffer registers for TCDRA and TCDRB. The TCBRA and TCBRB values are transferred to TCDRA and TCDRB with the transfer timing set in TMDR1. The initial value of TCBRA and TCBRB after a reset is H’FFFF.

16.2.29 Timer Dead Time Data Registers (TDDRA and TDDRB)

Address(es): MTU.TDDRA H’0_1000_1216, MTU.TDDRB H’0_1000_1A16

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TDDRA and TDDRB must not be accessed in eight bits; it should be accessed in 16 bits.

TDDRA and TDDRB are 16-bit readable/writable registers, used only in complementary PWM mode, that specify the MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) counter offset value. In complementary PWM mode, when the MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) counters are cleared and then restarted, the TDDRA (TDDRB) value is loaded into the MTU3.TCNT (MTU6.TCNT) counter and the count operation starts. The initial value of TDDRA and TDDRB after a reset is H’FFFF.

16.2.30 Timer Dead Time Enable Registers (TDERA and TDERB)

Address(es): MTU.TDERA H'0_1000_1234, MTU.TDERB H'0_1000_1A34

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDER
Initial Value	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)

Bit	Bit Name	Initial Value	R/W	Description
b0	TDER	1	R/(W)	Dead Time Enable 0: No dead time is generated 1: Dead time is generated*1
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. TDDRA and TDDRB must be set to 1 or a larger value.

TDERA and TDERB control dead time generation in complementary PWM mode. The MTU has one TDER each for MTU3 and MTU6. TDERA and TDERB should be modified only while TCNT stops.

TDER Bit (Dead Time Enable)

This bit specifies whether to generate dead time.

[Clearing condition]

- When 0 is written to TDER after reading TDER = 1

16.2.31 Timer Buffer Transfer Set Registers (TBTERA and TBTERB)

Address(es): MTU.TBTERA H'0_1000_1232, MTU.TBTERB H'0_1000_1A32

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	BTE[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	BTE[1:0]	All 0	R/W	Buffer Transfer Disable and Interrupt Skipping Link Setting These bits enable or disable transfer from the buffer registers*[1] used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping function 1. For details, see **Table 16.53**.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Applicable buffer registers (TBTERA):
MTU3.TGRC, MTU3.TGRD, MTU4.TGRC, MTU4.TGRD, and TCBRA
Applicable buffer registers (TBTERB):
MTU6.TGRC, MTU6.TGRD, MTU7.TGRC, MTU7.TGRD, and TCBRB

TBTERA and TBTERB enable or disable transfer from the buffer registers used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping 1 operation.

Table 16.53 Setting of BTE[1:0] Bits in TBTERA and TBTERB

Bit 1	Bit 0	
BTE1	BTE0	Description
0	0	Enables transfer from the buffer registers to the temporary registers*[1] and does not link the transfer with interrupt skipping function 1.
	1	Disables transfer from the buffer registers to the temporary registers.
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping function 1.*[2]
	1	Setting prohibited

Note 1. Data is transferred according to the MD3 to MD0 bit setting in TMDR1. For details, refer to **Section 16.3.8, Complementary PWM Mode**.

Note 2. When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits are cleared to 0 in the timer interrupt skipping set register (TITCR1A (TITCR1B)) or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), be sure to disable link of buffer transfer with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTERA (TBTERB)) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

16.2.32 Timer Waveform Control Registers (TWCRA and TWCRB)

Address(es): MTU.TWCRA H’0_1000_1260, MTU.TWCRB H’0_1000_1A60

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CCE	—	—	—	—	—	SCC	WRE
Initial Value	0*2	0	0	0	0	0	0	0
R/W	R/(W)	R/W	R/W	R/W	R/W	R/W	R/(W)	R/(W)*3

Bit	Bit Name	Initial Value	R/W	Description
b0	WRE	0	R/(W) *3	Waveform Retain Enable 0: Initial values specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are output 1: Initial output is inhibited
b1	SCC*1 *3	0	R/(W)	Synchronous Clearing Control (Only valid in register TWCRB) 0: Clearing of MTU6.TCNT and MTU7.TCNT in response to synchronous clearing for MTU0, MTU1, MTU2–MTU6, MTU7 is enabled. 1: Clearing of MTU6.TCNT and MTU7.TCNT in response to synchronous clearing for MTU0, MTU1, MTU2–MTU6, MTU7 is disabled.
b6 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b7	CCE*2	0	R/(W)	Compare Match Clear Enable 0: Counters are not cleared at MTU3.TGRA (MTU6.TGRA) compare match 1: Counters are cleared at MTU3.TGRA (MTU6.TGRA) compare match

Note 1. This bit is only valid in register TWCRB and is a reserved bit in register TWCRA.

Note 2. Do not set to 1 when complementary PWM mode 1 is not selected.

Note 3. Do not set to 1 when complementary PWM mode is not selected.

TWCRA and TWCRB control the output waveform when synchronous counter clearing occurs in MTU3.TNCT and MTU4.TNCT (MTU6.TNCT and MTU7.TNCT) in complementary PWM mode and specifies whether to clear the counters at MTU3.TGRA (MTU6.TGRA) compare match.

The CCE bit and WRE bit in TWCRA and TWCRB should be modified only while TCNT stops.

WRE Bit (Waveform Retain Enable)

This bit selects the waveform output when synchronous counter clearing occurs in complementary PWM mode. The initial output is inhibited with this function only when synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When synchronous clearing occurs outside this interval, the initial values

specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are output regardless of the WRE bit setting. The initial values specified in TOCR1A and TOCR2A (TOCR1B and TOCR2B) are also output when synchronous clearing occurs in the Tb interval at the trough immediately after MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) start operation.

For the Tb interval at the trough in complementary PWM mode, see **Figure 16.50**.

[Setting condition]

- When 1 is written to the WRE bit after reading WRE = 0

SCC Bit (Synchronous Clearing Control)

The setting of this bit selects whether MTU6.TCNT and MTU7.TCNT are or are not cleared when counter-synchronous clearing is generated for MTU0, MTU1, MTU2–MTU6, MTU7 in complementary PWM mode.

Make the complementary PWM mode settings for MTU6 and MTU7 when this function is in use. When writing a new value to the SCC bit while the counter is operating, do not change the values of the CCE and WRE bits.

Synchronous clearing from the MTU module only becomes disabled due to the setting of the SCC bit when synchronous clearing is generated outside the Tb interval in the trough. If synchronous clearing is generated within the Tb interval in the trough including immediately after the value at which MTU6.TCNT and MTU7.TCNT start, MTU6.TCNT and MTU7.TCNT are cleared.

Regarding the Tb interval in the trough in complementary PWM mode, see **Figure 16.50**.

[Setting condition]

- Writing of 1 to the SCC bit after reading it as 0

The corresponding bit in register TWCRA is reserved and is read as 0. When writing to TWCRA, write 0 to this bit.

CCE Bit (Compare Match Clear Enable)

This bit specifies whether to clear counters at MTU3.TGRA (MTU6.TGRA) compare match in complementary PWM mode.

[Setting condition]

When 1 is written to CCE after reading CCE = 0

16.2.33 Noise Filter Control Register n (NFCRn) (n = 0 to 4, 6, 7, 8, C)

(1) MTU0.NFCR0, MTU1.NFCR1, MTU2.NFCR2, MTU3.NFCR3, MTU4.NFCR4, MTU6.NFCR6, MTU7.NFCR7, MTU8.NFCR8

Address(es): MTU0.NFCR0 H'0_1000_1290, MTU1.NFCR1 H'0_1000_1291, MTU2.NFCR2 H'0_1000_1292, MTU3.NFCR3 H'0_1000_1293, MTU4.NFCR4 H'0_1000_1294, MTU6.NFCR6 H'0_1000_1A93, MTU7.NFCR7 H'0_1000_1A94, MTU8.NFCR8 H'0_1000_1298

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W*[1]	R/W*[1]	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	NFAEN	0	R/W	Noise Filter A Enable 0: The noise filter for the MTIOCnA pin is disabled. 1: The noise filter for the MTIOCnA pin is enabled.
b1	NFBEN	0	R/W	Noise Filter B Enable 0: The noise filter for the MTIOCnB pin is disabled. 1: The noise filter for the MTIOCnB pin is enabled.
b2	NFCEN	0	R/W*[1]	Noise Filter C Enable 0: The noise filter for the MTIOCnC pin is disabled. 1: The noise filter for the MTIOCnC pin is enabled.
b3	NFDEN	0	R/W*[1]	Noise Filter D Enable 0: The noise filter for the MTIOCnD pin is disabled. 1: The noise filter for the MTIOCnD pin is enabled.
b5, b4	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b5 b4 0 0: P0ϕ/1 0 1: P0ϕ/8 1 0: P0ϕ/32 1 1: Clock source for counting
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. These bits are reserved in the NFCR1 and NFCR2 registers. These bits are read as 0 and writing to them has no effect.

The NFCRn register (n = 0 to 4, 6, 7, or 8) specifies the noise filter function of the input capture input pin of the corresponding channel.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTIOCnA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTIOCnB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTIOCnC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTIOCnD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register or set the TMDR.MD[3:0] bits to a value other than that for normal mode (0000b) before doing so.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function. When the NFCS[1:0] bits are set to 11b, i.e. selecting the external clock as the source to drive counting, wait for two cycles of the external clock before setting the input capture function.

(2) MTU0.NFCRC

Address(es): MTU0.NFCRC H'0_1000_1299

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	NFAEN	0	R/W	Noise Filter A Enable 0: The noise filter for the MTCLKA pin is disabled. 1: The noise filter for the MTCLKA pin is enabled.
b1	NFBEN	0	R/W	Noise Filter B Enable 0: The noise filter for the MTCLKB pin is disabled. 1: The noise filter for the MTCLKB pin is enabled.
b2	NFCEN	0	R/W	Noise Filter C Enable 0: The noise filter for the MTCLKC pin is disabled. 1: The noise filter for the MTCLKC pin is enabled.
b3	NFDEN	0	R/W	Noise Filter D Enable 0: The noise filter for the MTCLKD pin is disabled. 1: The noise filter for the MTCLKD pin is enabled.
b5, b4	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b5 b4 0 0: P0ϕ/1 0 1: P0ϕ/2 1 0: P0ϕ/8 1 1: P0ϕ/32
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

The NFCRC register is used in common with all channels and specifies the noise filter function of the external clock pins.

NFAEN Bit (Noise Filter A Enable)

This bit disables or enables the noise filter for input from the MTCLKA pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFBEN Bit (Noise Filter B Enable)

This bit disables or enables the noise filter for input from the MTCLKB pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCEN Bit (Noise Filter C Enable)

This bit disables or enables the noise filter for input from the MTCLKC pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFDEN Bit (Noise Filter D Enable)

This bit disables or enables the noise filter for input from the MTCLKD pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, do so after stopping the internal counter.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. After setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval to set the input capture function.

16.2.34 Noise Filter Control Register 5 (NFCR5)

Address(es): MTU5.NFCR5 H'0_1000_1A95

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		—	NFWEN	NFVEN	NFUEN
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	NFUEN	0	R/W	Noise Filter U Enable 0: The noise filter for the MTIC5U pin is disabled. 1: The noise filter for the MTIC5U pin is enabled.
b1	NFVEN	0	R/W	Noise Filter V Enable 0: The noise filter for the MTIC5V pin is disabled. 1: The noise filter for the MTIC5V pin is enabled.
b2	NFWEN	0	R/W	Noise Filter W Enable 0: The noise filter for the MTIC5W pin is disabled. 1: The noise filter for the MTIC5W pin is enabled.
b3	—	0	R/W	Reserved These bits are read as 0. The write value should be 0.
b5, b4	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b5 b4 0 0: P0ϕ/1 0 1: P0ϕ/8 1 0: P0ϕ/32 1 1: Clock source for counting
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

NFUEN Bit (Noise Filter U Enable)

This bit disables or enables the noise filter for input from the MTIC5U pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFVEN Bit (Noise Filter V Enable)

This bit disables or enables the noise filter for input from the MTIC5V pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFWEN Bit (Noise Filter W Enable)

This bit disables or enables the noise filter for input from the MTIC5W pin. Since unexpected edges may be internally generated when the value of this bit is changed, select the output compare function for the relevant pin in the timer I/O control register.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits set the sampling interval for the noise filters. When setting the NFCS[1:0] bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

16.2.35 Timer A/D Converter Start Request Control Register (TADCR)

(1) MTU4.TADCR

Address(es): MTU4.TADCR H'0_1000_1240

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		—	—	—	—	—	—	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ITB4VE	0	R/W	TCIV4 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4BN and TCIV4 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4BN and TCIV4 interrupt skipping 1 are linked
b1	ITB3AE	0	R/W	TGIA3 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4BN and TGIA3 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4BN and TGIA3 interrupt skipping 1 are linked
b2	ITA4VE	0	R/W	TCIV4 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4AN and TCIV4 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4AN and TCIV4 interrupt skipping 1 are linked
b3	ITA3AE	0	R/W	TGIA3 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG4AN and TGIA3 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG4AN and TGIA3 interrupt skipping 1 are linked
b4	DT4BE	0	R/W	Down-Count TRG4BN Enable*3 0: A/D converter start requests (TRG4BN) disabled during MTU4.TCNT down-count operation 1: A/D converter start requests (TRG4BN) enabled during MTU4.TCNT down-count operation
b5	UT4BE	0	R/W	Up-Count TRG4BN Enable 0: A/D converter start requests (TRG4BN) disabled during MTU4.TCNT up-count operation 1: A/D converter start requests (TRG4BN) enabled during MTU4.TCNT up-count operation
b6	DT4AE	0	R/W	Down-Count TRG4AN Enable*3 0: A/D converter start requests (TRG4AN) disabled during MTU4.TCNT down-count operation 1: A/D converter start requests (TRG4AN) enabled during MTU4.TCNT down-count operation
b7	UT4AE	0	R/W	Up-Count TRG4AN Enable 0: A/D converter start requests (TRG4AN) disabled during MTU4.TCNT up-count operation 1: A/D converter up requests (TRG4AN) enabled during MTU4.TCNT down-count operation
b13 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b15, b14	BF[1:0]	All 0	R/W	MTU4.TADCOBRA/TADCOBRB Transfer Timing Select See **Table 16.54** for details. These bits specify the transfer timing from MTU4.TADCOBRA and MTU4.TADCOBRB to MTU4.TADCORA and MTU4.TADCORB.

Note: The TADCR register in MTU4 must not be accessed in eight bits; it should be accessed in 16 bits.

Note 1. Clear the bit to 0 when interrupt skipping is disabled (the T3AEN or T4VEN bit in TITCR1A is cleared to 0 or the T3ACOR or T4VCOR bits in TITCR1A are cleared to 0).

Note 2. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.

Note 3. Clear the bit to 0 when complementary PWM mode is not selected.

TADCR enables or disables A/D converter start requests and specifies whether to link A/D converter start requests with interrupt skipping function. The MTU has one TADCR each for MTU4 and MTU7.

Table 16.54 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU4)

Bit 15	Bit 14	Description			
BF[1]	BF[0]	Complementary PWM Mode	Reset-Synchronized PWM Mode	PWM Mode 1	Normal Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4).
0	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the crest of the TCNT count in MTU4.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU3.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU4.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) when a compare match occurs between TCNT and TGRA in MTU4.
1	0	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the trough of the TCNT count in MTU4.	Setting prohibited	Setting prohibited	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU4) to the cycle set register (TADCORA or TADCORB in MTU4) at the crest and trough of the TCNT count in MTU4.	Setting prohibited	Setting prohibited	Setting prohibited

(2) MTU7.TADCR

Address(es): MTU7.TADCR H'0_1000_1A40

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF[1:0]		—	—	—	—	—	—	UT7AE	DT7AE	UT7BE	DT7BE	ITA6AE	ITA7VE	ITB6AE	ITB7VE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ITB7VE	0	R/W	TCIV7 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7BN and TCIV7 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7BN and TCIV7 interrupt skipping 1 are linked
b1	ITB6AE	0	R/W	TGIA6 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7BN and TGIA6 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7BN and TGIA6 interrupt skipping 1 are linked
b2	ITA7VE	0	R/W	TCIV7 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7AN and TCIV7 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7AN and TCIV7 interrupt skipping 1 are linked
b3	ITA6AE	0	R/W	TGIA6 Interrupt Skipping Link Enable*1 *2 *3 0: A/D converter start request signal TRG7AN and TGIA6 interrupt skipping 1 are not linked 1: A/D converter start request signal TRG7AN and TGIA6 interrupt skipping 1 are linked
b4	DT7BE	0	R/W	Down-Count TRG7BN Enable*3 0: A/D converter start requests (TRG7BN) disabled during MTU7.TCNT down-count operation 1: A/D converter start requests (TRG7BN) enabled during MTU7.TCNT down-count operation
b5	UT7BE	0	R/W	Up-Count TRG7BN Enable 0: A/D converter start requests (TRG7BN) disabled during MTU7.TCNT up-count operation 1: A/D converter start requests (TRG7BN) enabled during MTU7.TCNT up-count operation
b6	DT7AE	0	R/W	Down-Count TRG7AN Enable*3 0: A/D converter start requests (TRG7AN) disabled during MTU7.TCNT down-count operation 1: A/D converter start requests (TRG7AN) enabled during MTU7.TCNT down-count operation
b7	UT7AE	0	R/W	Up-Count TRG7AN Enable 0: A/D converter start requests (TRG7AN) disabled during MTU7.TCNT up-count operation 1: A/D converter up requests (TRG7AN) enabled during MTU7.TCNT down-count operation
b13 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b15, b14	BF[1:0]	All 0	R/W	MTU7.TADCOBRA/TADCOBRB Transfer Timing Select See **Table 16.55** for details. These bits specify the transfer timing from MTU7.TADCOBRA and MTU7.TADCOBRB to MTU7.TADCORA and MTU7.TADCORB.

Note: The TADCR register in MTU7 must not be accessed in eight bits; it should be accessed in 16 bits.

Note 1. Clear the bit to 0 when interrupt skipping is disabled (the T6AEN or T7VEN bit in TITCR1B is cleared to 0 or the T6ACOR or T7VCOR bits in TITCR1B are cleared to 0).

Note 2. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.

Note 3. Clear the bit to 0 when complementary PWM mode is not selected.

Table 16.55 Setting of Transfer Timing by TADCR.BF[1:0] Bits (MTU7)

Bit 15	Bit 14	Description			
BF[1]	BF[0]	Complementary PWM Mode	Reset-Synchronized PWM Mode	PWM Mode 1	Normal Mode
0	0	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).	Does not transfer data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7).
0	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the crest of the TCNT count in MTU7.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU6.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU7.	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) when a compare match occurs between TCNT and TGRA in MTU7.
1	0	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the trough of the TCNT count in MTU7.	Setting prohibited	Setting prohibited	Setting prohibited
1	1	Transfers data from the cycle set buffer register (TADCOBRA or TADCOBRB in MTU7) to the cycle set register (TADCORA or TADCORB in MTU7) at the crest and trough of the TCNT count in MTU7.	Setting prohibited	Setting prohibited	Setting prohibited

16.2.36 Timer A/D Converter Start Request Cycle Set Registers (TADCORA and TADCORB)

Address(es): MTU4.TADCORA H’0_1000_1244, MTU4.TADCORB H’0_1000_1246, MTU7.TADCORA H’0_1000_1A44, MTU7.TADCORB H’0_1000_1A46

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TADCORA and TADCORB must not be accessed in eight bits; it should be accessed in 16 bits.

Note 1. When the A/D converter start request delaying function linked with skipping function 1 (for details, see **Section 16.3.9(5), A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1**) is used, the value of this register should be H’0002 to TCDRA setting - 2 in MTU4 and H’0002 to TCDRB setting - 2 in MTU7.

Note 2. When interrupt skipping function 2 is used and the difference between the TADCORA value and the TADCORB value is small, the skipping count may not be counted correctly and the A/D converter start request may not be generated with the expected timing in some cases. The TADCORA and TADCORB values should satisfy the following conditions.

(1) When skipping function 2 is specified with the skipping count set to 0

- The difference between the TADCORA and TADCORB values should be equal to or greater than 4.
- The TADCORA compare interval should be equal to or greater than 4 P0ϕ cycles (the TADCORA update value should be the previous value + 4 or greater, or previous value - 4 or smaller).
- The TADCORB compare interval should be equal to or greater than 4 P0ϕ cycles (the TADCORB update value should be the previous value + 4 or greater, or previous value - 4 or smaller).

(2) When skipping function 2 is specified with the skipping count set to 1 or greater

- The difference between the TADCORA and TADCORB values should be equal to or greater than 2.
- The TADCORB compare interval should be equal to or greater than 2 P0ϕ cycles (the TADCORB update value should be the previous value + 2 or greater, or previous value - 2 or smaller).

TADCORA and TADCORB are 16-bit readable/writable registers that issue a corresponding A/D converter start request when the MTUn.TCNT (n = 4, 7) count reaches the value in TADCORA or TADCORB.

MTUn.TADCORA and TADCORB are initialized to H’FFFF by a reset.

16.2.37 Timer A/D Converter Start Request Cycle Set Buffer Registers (TADCOBRA and TADCOBRB)

Address(es): MTU4.TADCORA H’0_1000_1248, MTU4.TADCORB H’0_1000_124A, MTU7.TADCORA H’0_1000_1A48, MTU7.TADCORB H’0_1000_1A4A

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TADCOBRA and TADCOBRB must not be accessed in eight bits; it should be accessed in 16 bits.

TADCOBRA and TADCOBRB are 16-bit readable/writable registers whose values are transferred to TADCORA and TADCORB, respectively, when the crest or trough of the MTUn.TCNT count is reached.

TADCOBRA and TADCOBRB are initialized to H’FFFF by a reset.

16.2.38 Timer Interrupt Skipping Mode Registers (TITMRA and TITMRB)

Address(es): MTU.TITMRA H’0_1000_123A, MTU.TITMRB H’0_1000_1A3A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TITM
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TITM	0	R/W	Interrupt Skipping Function Select Selects one of the two types of interrupt skipping functions. 0: Selects interrupt skipping function 1*1 1: Selects interrupt skipping function 2*2
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. TITCR1A or TITCR1B is used to enable interrupt skipping function 1.

Note 2. TITCR2A or TITCR2B is used to enable interrupt skipping function 2.

TITMRA and TITMRB are used to select either of two skipping functions for the TITMRA and TITMRB registers.

16.2.39 Timer Interrupt Skipping Set Registers 1 (TITCR1A and TITCR1B)

(1) MTU.TITCR1A

Address(es): MTU.TITMRA H'0_1000_1230

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	T3AEN	T3ACOR[2:0]			T4VEN	T4VCOR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T4VCOR[2:0]	All 0	R/W	TCIV4 Interrupt Skipping Count Setting These bits specify the TCIV4 interrupt skipping count within the range from 0 to 7.*1 For details, see **Table 16.56**.
b3	T4VEN	0	R/W	T4VEN 0: TCIV4 interrupt skipping disabled 1: TCIV4 interrupt skipping enabled
b6 to b4	T3ACOR[2:0]	All 0	R/W	TGIA3 Interrupt Skipping Count Setting These bits specify the TGIA3 interrupt skipping count within the range from 0 to 7. For details, see **Table 16.57**.
b7	T3AEN	0	R/W	T3AEN 0: TGIA3 interrupt skipping disabled 1: TGIA3 interrupt skipping enabled

Note 1. When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed.
Before changing the interrupt skipping count, be sure to clear the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0 to clear the skipping counter (TITCNT1A).

(2) MTU.TITCR1B

Address(es): MTU.TITCR1B H'0_1000_1A30

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	T6AEN	T6ACOR[2:0]			T7VEN	T7VCOR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T7VCOR[2:0]	All 0	R/W	TCIV7 Interrupt Skipping Count Setting These bits specify the TCIV7 interrupt skipping count within the range from 0 to 7.*[1] For details, see **Table 16.58**.
b3	T7VEN	0	R/W	T7VEN 0: TCIV7 interrupt skipping disabled 1: TCIV7 interrupt skipping enabled
b6 to b4	T6ACOR[2:0]	All 0	R/W	TGIA6 Interrupt Skipping Count Setting These bits specify the TGIA6 interrupt skipping count within the range from 0 to 7.*[1] For details, see **Table 16.59**.
b7	T6AEN	0	R/W	T6AEN 0: TGIA6 interrupt skipping disabled 1: TGIA6 interrupt skipping enabled

Note 1. When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed.
Before changing the interrupt skipping count, be sure to clear the TITCR1B.T6AEN and TITCR1B.T7VEN bits to 0 to clear the skipping counter (TITCNT1B).

TITCR1A and TITCR1B enable or disable interrupt skipping and specify the interrupt skipping count.

The setting in the TITCR1A (TITCR1B) register is valid only while the TITM bit in TITMRA (TITMRB) is set to 0; when the TITM bit in TITMRA (TITMRB) is set to 1, the setting in TITCR1A (TITCR1B) is cleared.

Table 16.56 Setting of Interrupt Skipping Count by T4VCOR[2:0] Bits

Bit 2	Bit 1	Bit 0	
T4VCOR2	T4VCOR1	T4VCOR0	Description
0	0	0	Does not skip TCIV4 interrupts.
		1	Sets the TCIV4 interrupt skipping count to 1.
	1	0	Sets the TCIV4 interrupt skipping count to 2.
		1	Sets the TCIV4 interrupt skipping count to 3.
1	0	0	Sets the TCIV4 interrupt skipping count to 4.
		1	Sets the TCIV4 interrupt skipping count to 5.
	1	0	Sets the TCIV4 interrupt skipping count to 6.
		1	Sets the TCIV4 interrupt skipping count to 7.

Table 16.57 Setting of Interrupt Skipping Count by T3ACOR[2:0] Bits

Bit 6	Bit 5	Bit 4	
T3ACOR2	T3ACOR1	T3ACOR0	Description
0	0	0	Does not skip TGIA3 interrupts.
		1	Sets the TGIA3 interrupt skipping count to 1.
	1	0	Sets the TGIA3 interrupt skipping count to 2.
		1	Sets the TGIA3 interrupt skipping count to 3.
1	0	0	Sets the TGIA3 interrupt skipping count to 4.
		1	Sets the TGIA3 interrupt skipping count to 5.
	1	0	Sets the TGIA3 interrupt skipping count to 6.
		1	Sets the TGIA3 interrupt skipping count to 7.

Table 16.58 Setting of Interrupt Skipping Count by T7VCOR[2:0] Bits

Bit 2	Bit 1	Bit 0	
T7VCOR2	T7VCOR1	T7VCOR0	Description
0	0	0	Does not skip TCIV7 interrupts.
		1	Sets the TCIV7 interrupt skipping count to 1.
	1	0	Sets the TCIV7 interrupt skipping count to 2.
		1	Sets the TCIV7 interrupt skipping count to 3.
1	0	0	Sets the TCIV7 interrupt skipping count to 4.
		1	Sets the TCIV7 interrupt skipping count to 5.
	1	0	Sets the TCIV7 interrupt skipping count to 6.
		1	Sets the TCIV7 interrupt skipping count to 7.

Table 16.59 Setting of Interrupt Skipping Count by T6ACOR[2:0] Bits

Bit 6	Bit 5	Bit 4	
T6ACOR2	T6ACOR1	T6ACOR0	Description
0	0	0	Does not skip TGIA6 interrupts.
		1	Sets the TGIA6 interrupt skipping count to 1.
	1	0	Sets the TGIA6 interrupt skipping count to 2.
		1	Sets the TGIA6 interrupt skipping count to 3.
1	0	0	Sets the TGIA6 interrupt skipping count to 4.
		1	Sets the TGIA6 interrupt skipping count to 5.
	1	0	Sets the TGIA6 interrupt skipping count to 6.
		1	Sets the TGIA6 interrupt skipping count to 7.

16.2.40 Timer Interrupt Skipping Counters 1 (TITCNT1A and TITCNT1B)

(1) MTU.TITCNT1A

Address(es): MTU.TITCNT1A H'0_1000_1231

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	T3ACNT[2:0]			—	T4VCNT[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T4VCNT[2:0]	All 0	R	TCIV4 Interrupt Counter While the T4VEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TCIV4 interrupt occurs.
b3	—	0	R	Reserved This bit is read as 0.
b6 to b4	T3ACNT[2:0]	All 0	R	TGIA3 Interrupt Counter While the T3AEN bit in TITCR1A is set to 1, the count in these bits is incremented every time a TGIA3 interrupt occurs.
b7	—	0	R	Reserved This bit is read as 0.

Note: To clear the TITCNT1A, clear the TITCR1A.T3AEN and TITCR1A.T4VEN bits to 0.

TITCNT1A and TITCNT1B are 8-bit readable/writable counters. TITCNTA and TITCNTB retain their values even after stopping the count operation of MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT).

T4VCNT[2:0] Bits (TCIV4 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T4VEN bit in TITCR1A is cleared to 0
- When the T4VCOR[2:0] bits in TITCR1A are cleared to 000b
- When the T4VCNT[2:0] bits in TITCNT1A match the T4VCOR[2:0] bits in TITCR1A

T3ACNT[2:0] Bits (TGIA3 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRA is 1
- When the T3AEN bit in TITCR1A is cleared to 0
- When the T3ACOR[2:0] bits in TITCR1A are cleared to 000b
- When the T3ACNT[2:0] bits in TITCNT1A match the T3ACOR[2:0] bits in TITCR1A

(2) MTU.TITCNT1B

Address(es): MTU.TITCNT1B H'0_1000_1A31

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	T6ACNT[2:0]			—	T7VCNT[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	T7VCNT[2:0]	All 0	R	TCIV7 Interrupt Counter While the T7VEN bit in TITCR1B is set to 1, the count in these bits is incremented every time a TCIV7 interrupt occurs.
b3	—	0	R	Reserved This bit is read as 0.
b6 to b4	T6ACNT[2:0]	All 0	R	TGIA6 Interrupt Counter While the T6AEN bit in TITCR1B is set to 1, the count in these bits is incremented every time a TGIA6 interrupt occurs.
b7	—	0	R	Reserved This bit is read as 0.

Note: To clear the TITCNT1B, clear the TITCR1B.T6AEN and TITCR1B.T7VEN bits to 0.

T7VCNT[2:0] Bits (TCIV7 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRB is 1
- When the T7VEN bit in TITCR1B is cleared to 0
- When the T7VCOR[2:0] bits in TITCR1B are cleared to 000b
- When the T7VCNT[2:0] bits in TITCNT1B match the T7VCOR[2:0] bits in TITCR1B

T6ACNT[2:0] Bits (TGIA6 Interrupt Counter)

[Clearing conditions]

- When the TITM bit in TITMRB is 1
- When the T6AEN bit in TITCR1B is cleared to 0
- When the T6ACOR[2:0] bits in TITCR1B are cleared to 000b
- When the T6ACNT[2:0] bits in TITCNT1B match the T6ACOR[2:0] bits in TITCR1B

16.2.41 Timer Interrupt Skipping Set Registers 2 (TITCR2A and TITCR2B)

(1) MTU.TITCR2A

Address(es): MTU.TITCR2A H'0_1000_123B

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4COR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG4COR [2:0]	All 0	R/W	TRG4AN/TRG4BN Interrupt Skipping Count Setting These bits specify the TRG4AN/TRG4BN interrupt skipping count within the range from 0 to 7. For details, see **Table 16.60**.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

TITCR2A and TITCR2B specify the interrupt skipping count for TRG4AN and TRG4BN (TRG7AN and TRG7BN). This setting is valid only while TITMRA or TITMRB is set to 1.

Table 16.60 Setting of Interrupt Skipping Count by TRG4COR[2:0] Bits

Bit 2	Bit 1	Bit 0	
TRG4COR2	TRG4COR1	TRG4COR0	Description
0	0	0	Does not skip TRG4AN and TRG4BN interrupts.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 1.
	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 2.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 3.
1	0	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 4.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 5.
	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 6.
		1	Sets the TRG4AN and TRG4BN interrupt skipping count to 7.

(2) MTU.TITCR2B

Address(es): MTU.TITCR2B H'0_1000_1A3B

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7COR[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG7COR [2:0]	All 0	R/W	TRG7AN/TRG7BN Interrupt Skipping Count Setting These bits specify the TRG7AN/TRG7BN interrupt skipping count within the range from 0 to 7. For details, see **Table 16.61**.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Table 16.61 Setting of Interrupt Skipping Count by TRG7COR[2:0] Bits

Bit 2	Bit 1	Bit 0	
TRG7COR2	TRG7COR1	TRG7COR0	Description
0	0	0	Does not skip TRG7AN and TRG7BN interrupts.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 1.
	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 2.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 3.
1	0	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 4.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 5.
	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 6.
		1	Sets the TRG7AN and TRG7BN interrupt skipping count to 7.

16.2.42 Timer Interrupt Skipping Counters 2 (TITCNT2A and TITCNT2B)

(1) MTU.TITCNT2A

Address(es): MTU.TITCNT2A H'0_1000_123C

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4CNT[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG4CNT [2:0]	All 0	R	TRG4AN/TRG4BN Interrupt Counter These bits start counting from the value set in TRG4COR[2:0] and the count decrements every time TRG4AN or TRG4BN is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.
b7 to b3	—	All 0	R	Reserved These bits are read as 0.

TITCNT2A and TITCNT2B start counting from the values set in the TRG4COR[2:0] and TRG7COR[2:0] bits and the count decrements every time TRG4AN or TRG4BN (TITCNT2A) is generated or TRG7AN or TRG7BN (TITCNT2B) is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts or the TRG7AN and TRG7BN interrupts become valid.

TRG4CNT[2:0] Bits (TRG4AN/TRG4BN Interrupt Counter)

These bits start counting from the value set in the TRG4COR[2:0] bits and the count decrements every time a TRG4AN or TRG4BN interrupt is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid.

[Clearing conditions]

- When the TITM bit in TITMRA is 0
- When the TRG4COR[2:0] bits in TITCR2A are cleared to 000b
- When the count of TRG4AN and TRG4BN occurrence matches the TRG4COR[2:0] value in TITCR2A

(2) MTU.TITCNT2B

Address(es): MTU.TITCNT2B H'0_1000_1A3C

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7CNT[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	TRG7CNT [2:0]	All 0	R	TRG7AN/TRG7BN Interrupt Counter These bits start counting from the value set in TRG7COR[2:0] and the count decrements every time TRG7AN or TRG7BN is generated. When the count reaches 0 and is reloaded, the TRG7AN and TRG7BN interrupts become valid.
b7 to b3	—	All 0	R	Reserved These bits are read as 0.

TRG7CNT[2:0] Bits (TRG7AN/TRG7BN Interrupt Counter)

These bits start counting from the value set in the TRG7COR[2:0] bits and the count decrements every time a TRG7AN or TRG7BN interrupt is generated. When the count reaches 0 and is reloaded, the TRG7AN and TRG7BN interrupts become valid.

[Clearing conditions]

- When the TITM bit in TITMRB is 0
- When the TRG7COR[2:0] bits in TITCR2B are cleared to 000b
- When the count of TRG7AN and TRG7BN occurrence matches the TRG7COR[2:0] value in TITCR2B

16.2.43 Bus Master Interface

The timer counter (MTU8.TCNT), general registers (MTU8.TGRn) for MTU8, and MTU1.TCNTLW, MTU1.TGRALW, and MTU1.TGRBLW registers when TMDR3.LWA = 1 are 32-bit registers. A 32-bit data bus to the bus master enables 32-bit read/write access. 8- and 16-bit read/write are not allowed. Access these registers in 32-bit units.

Excluding MTU8, the timer counters (MTU0.TCNT to MTU7.TCNT), general registers (MTU0.TGRn to MTU7.TGRn), timer subcounters (TCNTSA and TCNTSB), timer cycle buffer registers (TCBRA and TCBRB), timer dead time data registers (TDDRA and TDDRB), timer cycle data registers (TCDRA and TCDRB), timer A/D converter start request control registers (MTU4.TADCR and MTU7.TADCR), timer A/D converter start request cycle set registers (MTU4.TADCORA, MTU4.TADCORB, MTU7.TADCORA, and MTU7.TADCORB), and timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA, MTU4.TADCOBRB, MTU7.TADCOBRA, and MTU7.TADCOBRB) are 16-bit registers. A 16-bit data bus to the bus master enables 16-bit read/write access. 8-bit read/ write is not allowed. Access the registers in 16-bit units.

All registers other than the above registers are 8-bit registers. Read from/write to these registers in 8-bit units.

16.3 Operation

16.3.1 Basic Functions

Each channel has TCNT and TGR. TCNT in each channel performs up-counting and is also capable of free-running count, periodic count, and external event count operations.

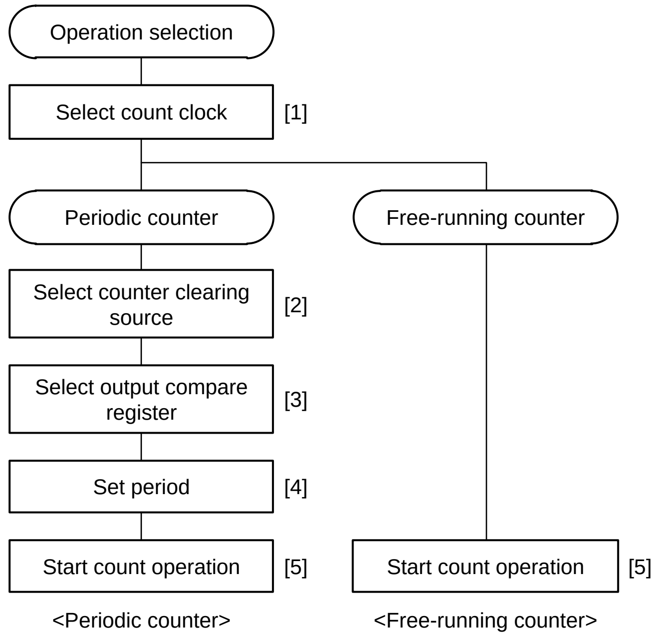
Each TGR can be used as an input capture register or an output compare register.

(1) Counter Operation

When one of bits CST0 to CST4 and CST8 in TSTRA, bits CST6 and CST7 in TSTRB, and bits CSTU5, CSTV5, and CSTW5 in MTU5.TSTR is set to 1, TCNT for the corresponding channel begins counting. TCNT can perform various count operations such as free-running count or periodic count operations.

(a) Example of Count Operation Setting Procedure

Figure 16.5 shows an example of the count operation setting procedure.



[1] Select the count clock source with bits TPSC[2:0] in TCR. At the same time, select the count clock source edge with bits CKEG[1:0] in TCR.

[2] For periodic counter operation, select the TGR to be used as the TCNT clearing source with bits CCLR[2:0] in TCR.

[3] Designate the TGR selected in [2] as an output compare register by means of TIOR.

[4] Set the periodic counter cycle in the TGR selected in [2].

[5] Set the CSTn bit in TSTRA, TSTRB, or MTU5.TSTR to 1 to start the counter operation.

Figure 16.5 Example of Count Operation Setting Procedure

(b) Free-Running Count Operation and Periodic Count Operation

Immediately after a reset, all TCNT counters are designated as free-running counters. When the CSTn bit in TSTRA, TSTRB, or TSTR in MTU5 is set to 1, the corresponding TCNT counter starts up-counting as a free-running counter. When TCNT overflows (from H'FFFF to H'0000), if the corresponding TIER.TCIEV bit is 1, an interrupt request is issued to the CPU. After an overflow, TCNT starts counting up again from H'0000.

Figure 16.6 illustrates free-running counter operation.

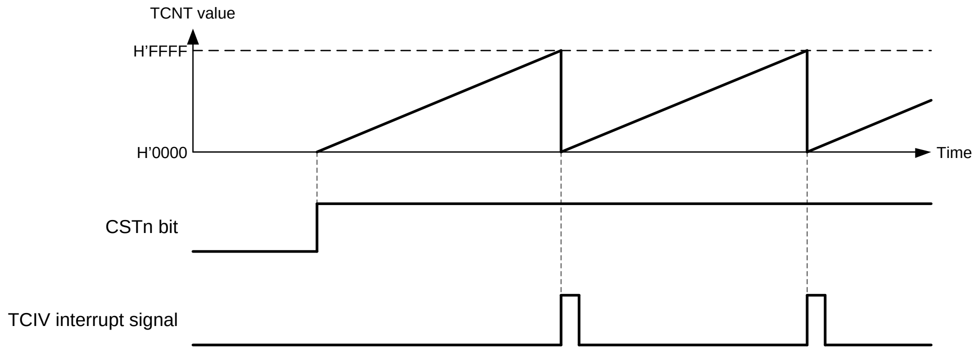


Figure 16.6 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, TCNT for the relevant channel performs periodic count operation. TGR for setting the cycle is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR[2:0] in TCR. After the settings have been made, TCNT starts up-counting as a periodic counter when the corresponding CSTn bit in TSTRA, TSTRB, or TSTR in MTU5 is set to 1. When the count matches the value in TGR, TCNT is cleared to H'0000.

If the value of the corresponding TIER.TGIE bit is 1 at this point, an interrupt request is issued to the CPU. After a compare match, TCNT starts counting up again from H'0000.

Figure 16.7 illustrates periodic counter operation.

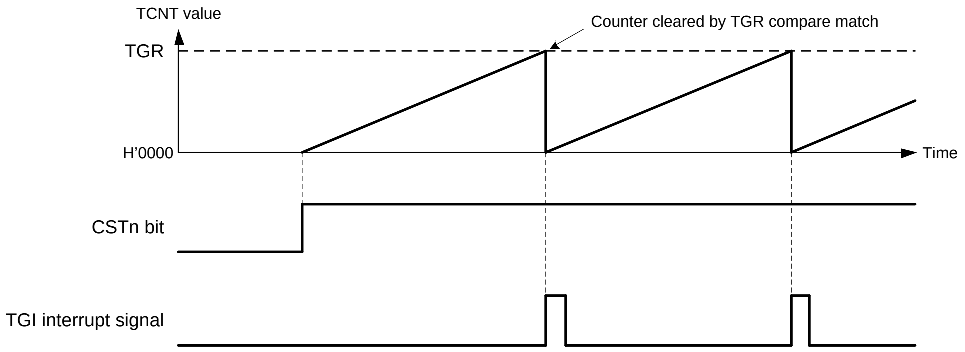


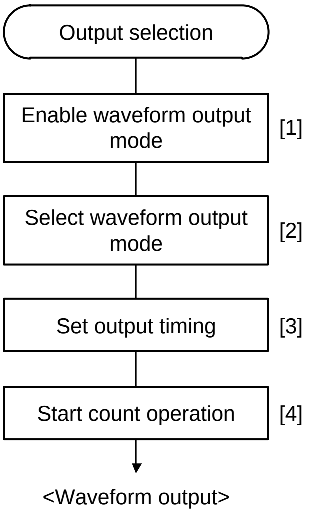
Figure 16.7 Periodic Counter Operation

(2) Waveform Output by Compare Match

Upon compare match, low, high, or toggle output from the corresponding pin can be performed. The compare match output operation is not available in MTU5.

(a) Example of Procedure for Setting Waveform Output by Compare Match

Figure 16.8 shows an example of the procedure for setting waveform output by compare match



[1] Enable TOERA output when outputting a waveform from the MTIOC pin of MTU3 and MTU4. Enable TOERB output when outputting a waveform from the MTIOC pin of MTU6 and MTU7.

[2] Select initial value low output or high output, and compare match output value low output, high output, or toggle output, by means of TIOR. The set initial value is output at the MTIOC pin until the first compare match occurs.

[3] Set the timing for compare match generation in TGR.

[4] Set the CSTn bit in TSTRA or TSTRB to 1 to start the count operation.

Figure 16.8 Example of Procedure for Setting Waveform Output by Compare Match

(b) Examples of Waveform Output Operation

Figure 16.9 shows an example of low output and high output.

In this example, TCNT has been designated as a free-running counter, and settings have been made so that high is output by compare match A and low is output by compare match B. When the pin level is the same as the specified level, the pin level does not change.

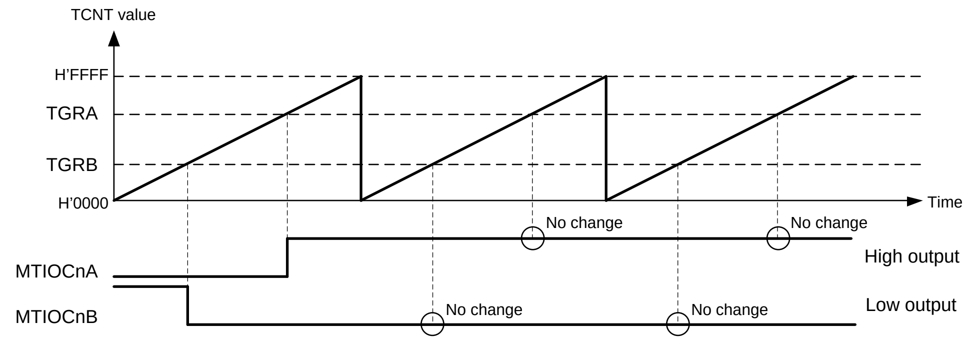


Figure 16.9 Example of low output and high output Operation (n = 0 to 4, 6, 7, 8)

Figure 16.10 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made so that the output is toggled by both compare match A and compare match B.

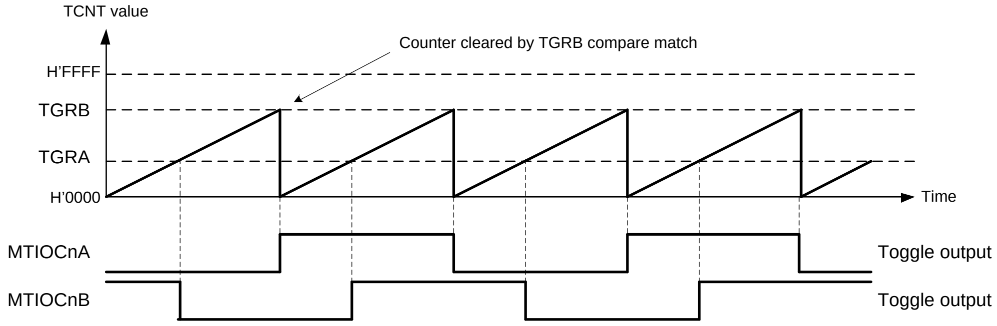


Figure 16.10 Example of Toggle Output Operation (n = 0 to 4, 6, 7, 8)

(3) Input Capture Function

The TCNT value can be transferred to TGR on detection of the MTIOCnm, MTIC5U, MTIC5V, or MTIC5W pin (n = 0 to 4, 6, 7, 8; m = A to D), input edge.

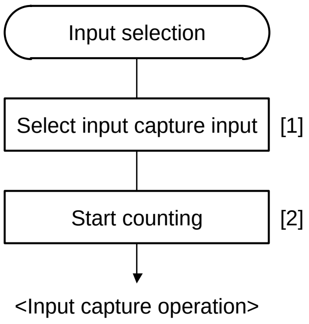
The rising edge, falling edge, or both edges can be selected as the detection edge. For MTU0 and MTU1, another channel’s count clock or compare match signal can also be specified as the input capture source.

NOTE

When another channel’s count clock is used as the input capture input for MTU0 and MTU1, P0ϕ/1 should not be selected as the count clock used for input capture input. Input capture will not be generated if P0ϕ/1 is selected.

(a) Example of Input Capture Operation Setting Procedure

Figure 16.11 shows an example of the input capture operation setting procedure.



[1] Designate TGR as an input capture register by means of TIOR, and select rising edge, falling edge, or both edges as the input capture source and input signal edge.

[2] Set the CSTn bit in TSTRA, TSTRB or MTU5.TSTR to 1 to start the count operation.

Figure 16.11 Example of Input Capture Operation Setting Procedure

(b) Example of Input Capture Operation

Figure 16.12 shows an example of input capture operation.

In this example, both rising and falling edges have been selected as the MTIOCnA pin input capture input edge, the falling edge has been selected as the MTIOCnB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT. (n = 0 to 4, 6, 7, 8)

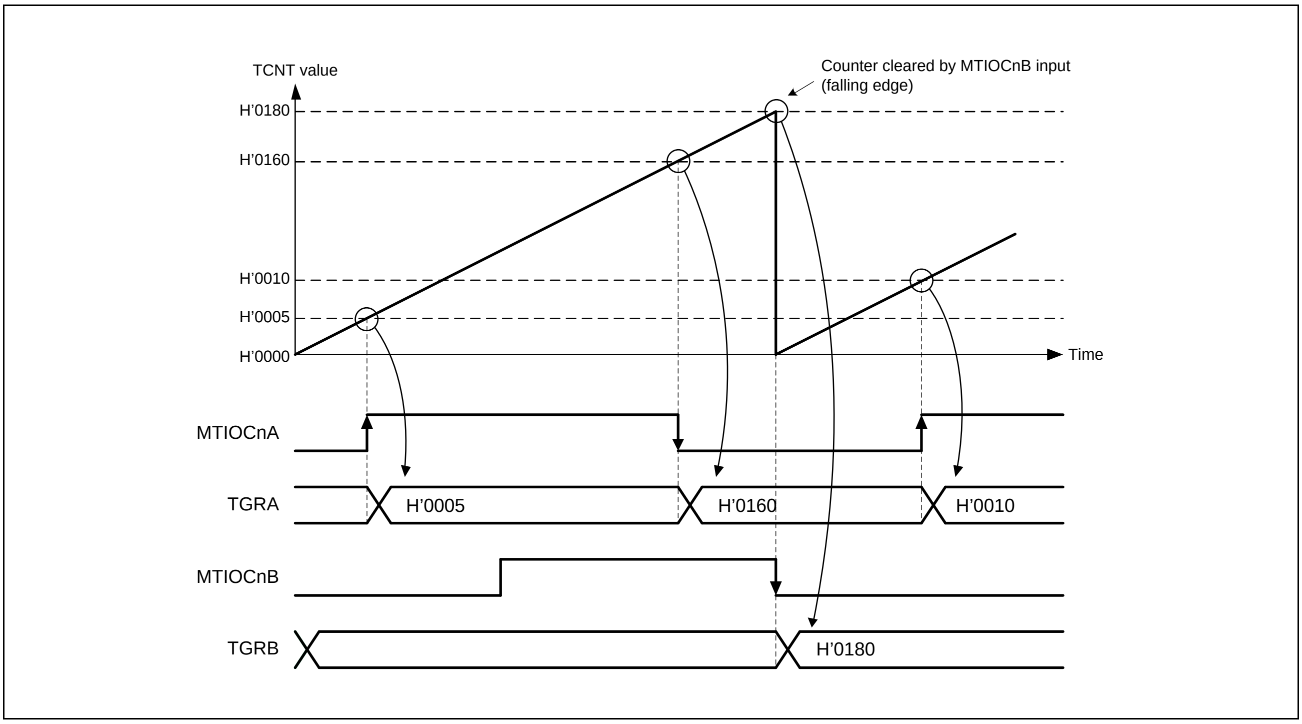


Figure 16.12 Example of Input Capture Operation (n = 0 to 4, 6, 7, 8)

16.3.2 Synchronous Operation

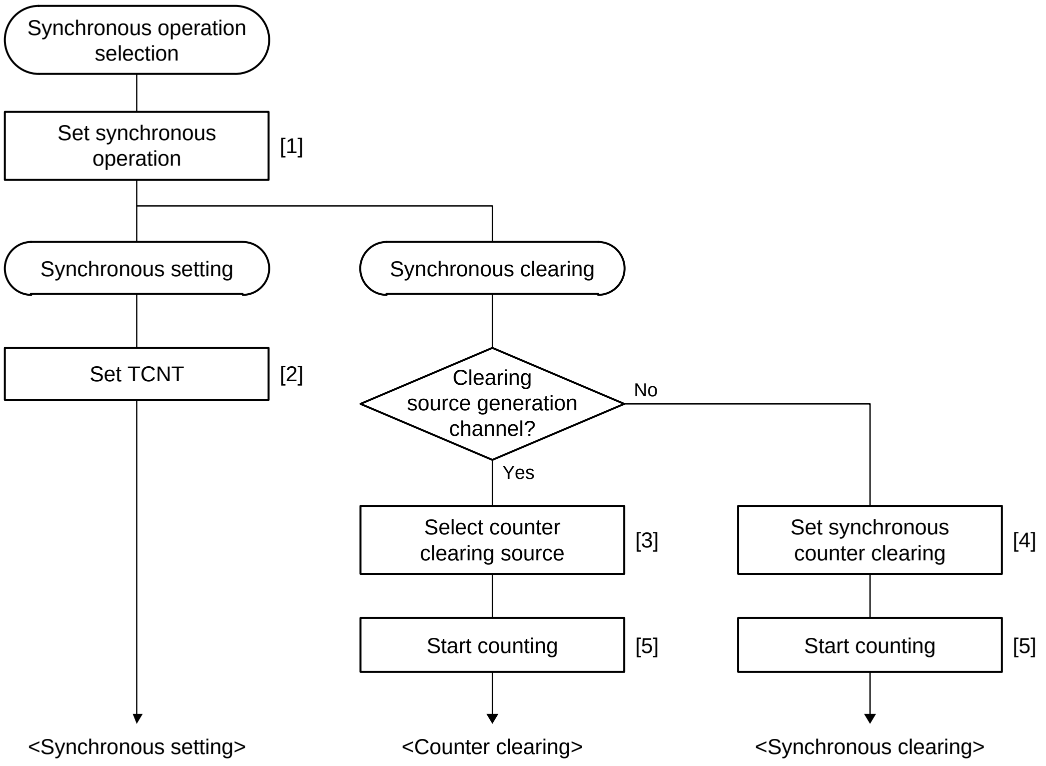
In synchronous operation, the values in multiple TCNT counters can be modified simultaneously (synchronous setting). In addition, multiple TCNT counters can be cleared simultaneously (synchronous clearing) by making the appropriate setting in TCR.

Synchronous operation can increase the number of TGR registers assigned to the same time base.

MTU0 to MTU4, MTU6, and MTU7 can all be designated for synchronous operation. MTU5 and MTU8 cannot be used for synchronous operation.

(1) Example of Synchronous Operation Setting Procedure

Figure 16.13 shows an example of the synchronous operation setting procedure.



[1] Set to 1 in the SYNC bits in TSYRA and TSYRB corresponding to the channels to be designated for synchronous operation.
[2] When the TCNT counter of any of the channels designated for synchronous operation is written to, the same value is simultaneously written to the other TCNT counters.
[3] Use the CCLR[2:0] bits in TCR to specify TCNT clearing by input capture/output compare, etc.
[4] Use the CCLR[2:0] bits in TCR to designate synchronous clearing for the counter clearing source.
[5] Set to 1 in the CSTn bits in TSTRA and TSTRB for the relevant channels, to start the count operation.

Figure 16.13 Example of Synchronous Operation Setting Procedure

(2) Example of Synchronous Operation

Figure 16.14 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for MTU0 to MTU 2, MTU0.TGRB compare match has been set as the counter clearing source in MTU0, and synchronous clearing has been set for the counter clearing source in MTU1 and MTU2.

Three-phase PWM waveforms are output from pins MTIOC0A, MTIOC1A, and MTIOC2A. At this time, synchronous setting and synchronous clearing by MTU0.TGRB compare match are performed for the TCNT counters in MTU0 to MTU2, and the data set in MTU0.TGRB is used as the PWM cycle.

For details of PWM modes, see **Section 16.3.5, PWM Modes**.

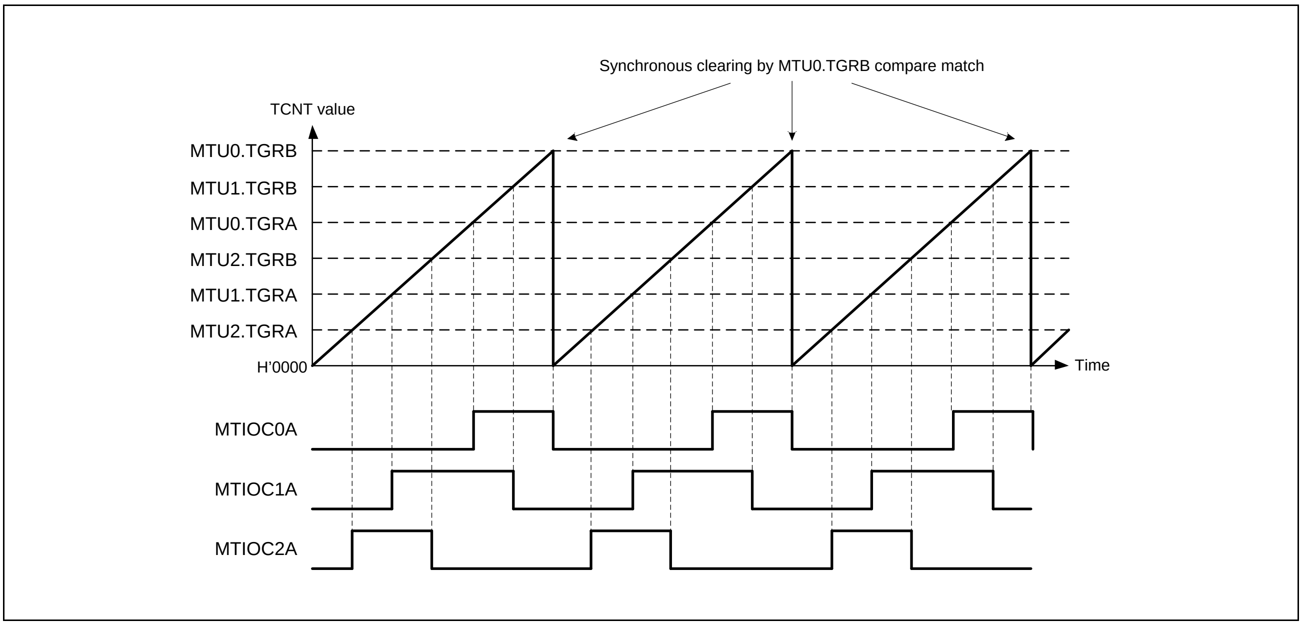


Figure 16.14 Example of Synchronous Operation

16.3.3 Buffer Operation

Buffer operation, provided for MTU0, MTU3, MTU4, MTU6, MTU7, and MTU8, enables TGRC and TGRD to be used as buffer registers. In MTU0, TGRF can also be used as a buffer register.

Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

NOTE

MTU0.TGRE cannot be designated as an input capture register and can only operate as a compare match register.

Table 16.62 shows the register combinations used in buffer operation.

Table 16.62 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
MTU0	TGRA	TGRC
	TGRB	TGRD
	TGRE	TGRF
MTU3	TGRA	TGRC
	TGRB	TGRD
MTU4	TGRA	TGRC
	TGRB	TGRD
MTU6	TGRA	TGRC
	TGRB	TGRD
MTU7	TGRA	TGRC
	TGRB	TGRD
MTU8	TGRA	TGRC
	TGRB	TGRD

When TGR is an output compare register

When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

This operation is illustrated in **Figure 16.15**.

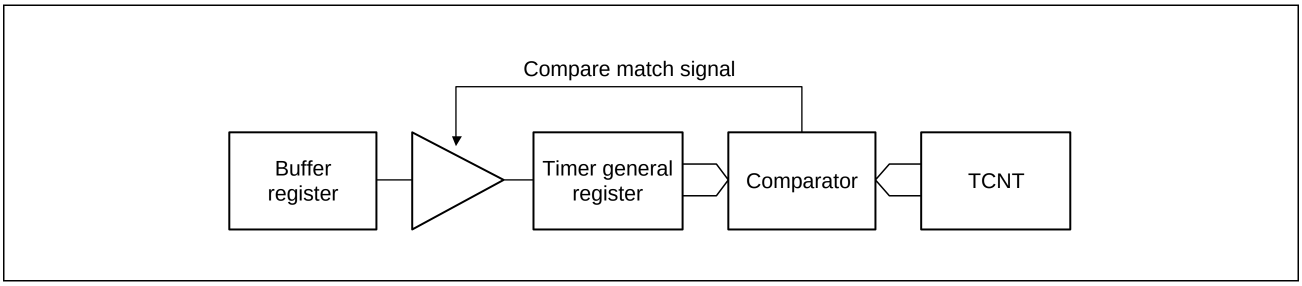


Figure 16.15 Compare Match Buffer Operation

When TGR is an input capture register

When an input capture occurs, the value in TCNT is transferred to TGR and the value previously held in TGR is transferred to the buffer register.

This operation is illustrated in **Figure 16.16**.

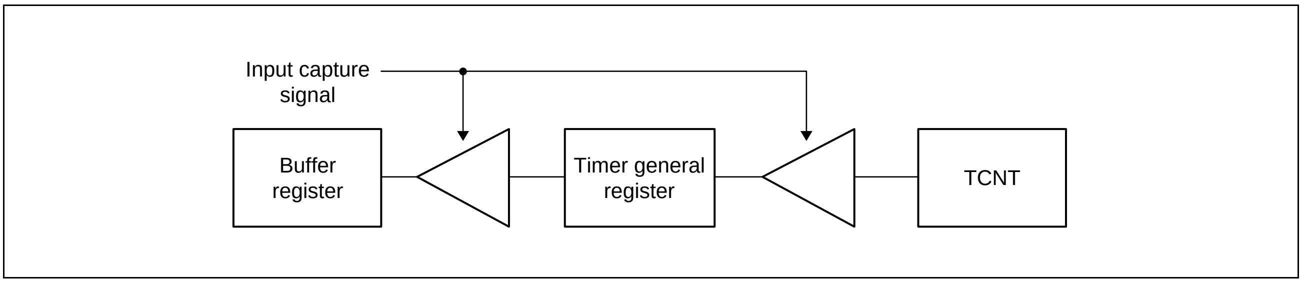
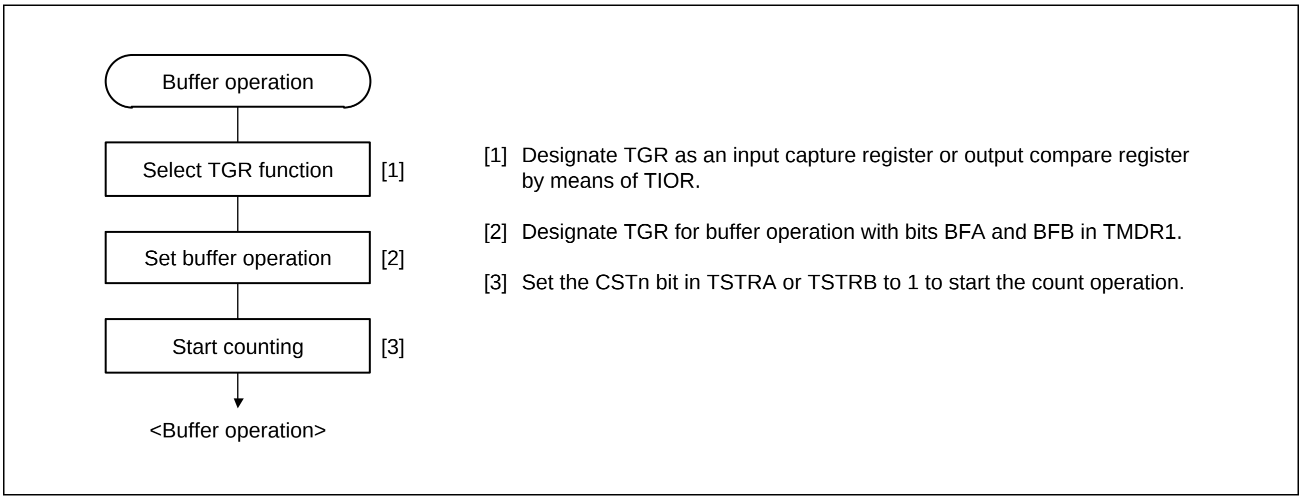


Figure 16.16 Input Capture Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 16.17 shows an example of the buffer operation setting procedure.



[1] Designate TGR as an input capture register or output compare register by means of TIOR.

[2] Designate TGR for buffer operation with bits BFA and BFB in TMDR1.

[3] Set the CSTn bit in TSTRA or TSTRB to 1 to start the count operation.

Figure 16.17 Example of Buffer Operation Setting Procedure

(2) Examples of Buffer Operation

(a) When TGR is an Output Compare Register

Figure 16.18 shows an operation example in which PWM mode 1 has been designated for MTU0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. In this example, the TTSA bit in TBTM is cleared to 0.

As buffer operation has been set, when compare match A occurs, the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time compare match A occurs.

For details of PWM modes, see **Section 16.3.5, PWM Modes**.

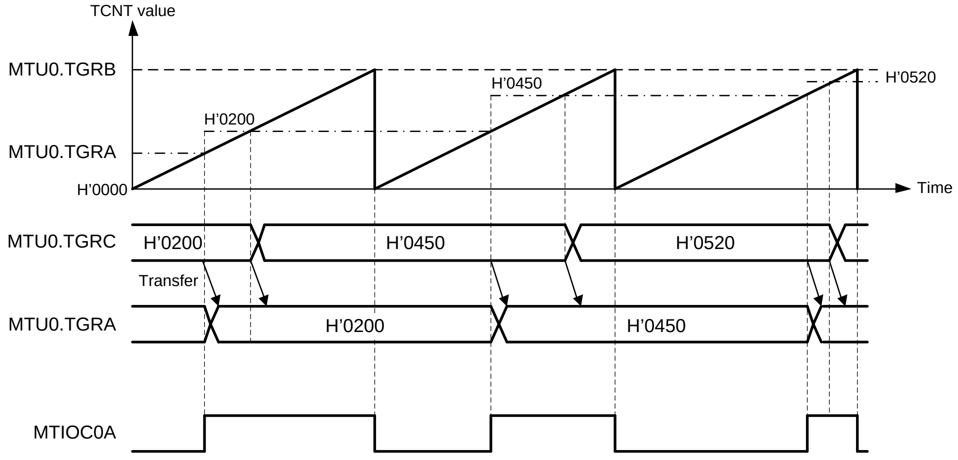


Figure 16.18 Example of Buffer Operation (1)

(b) When TGR is an Input Capture Register

Figure 16.19 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.

Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the MTIOCnA pin input capture input edge. (n = 0 to 4, 6, 7, 8)

As buffer operation has been set, when the TCNT value is transferred to TGRA upon occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

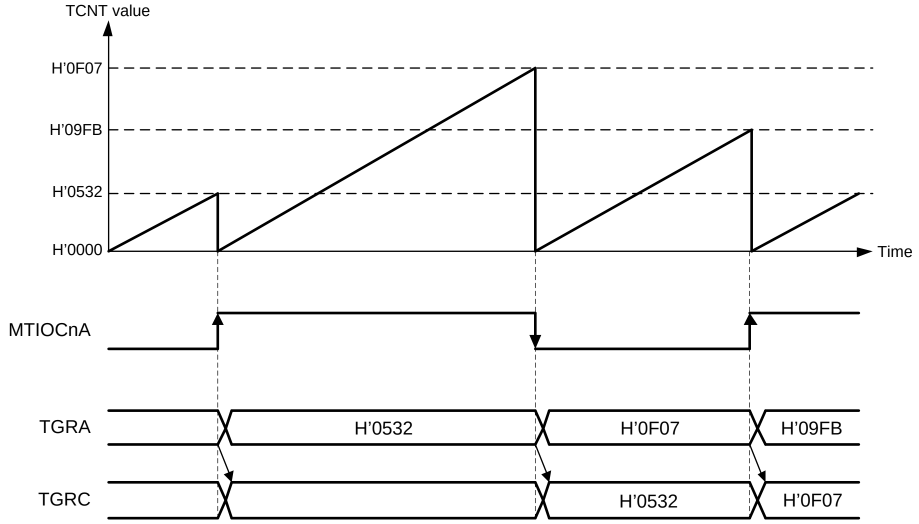


Figure 16.19 Example of Buffer Operation (2) (n = 0 to 4, 6, 7, 8)

(3) Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation

The timing for transfer from buffer registers to timer general registers can be selected in PWM mode 1 or 2 for MTU0 or in PWM mode 1 for MTU3, MTU4, MTU6, and MTU7 by setting the buffer operation transfer mode registers (MTUn.TBTM (n = 0, 3, 4, 6, 7)). Either compare match (value after reset) or TCNT clearing can be selected for the transfer timing. TCNT clearing as transfer timing is one of the following cases.

- When TCNT overflows (H’FFFF to H’0000)
- When H’0000 is written to TCNT during counting
- When TCNT is cleared to H’0000 under the condition specified in the CCLR[2:0] bits in TCR

NOTE

TBTM must be modified only while TCNT stops.

Figure 16.20 shows an operation example in which PWM mode 1 is designated for MTU0 and buffer operation is designated for MTU0.TGRA and MTU0.TGRC. The settings used in this example are MTU0.TCNT clearing by compare match B, high output at compare match A, and low output at compare match B. The TTSA bit in MTU0.TBTM is set to 1.

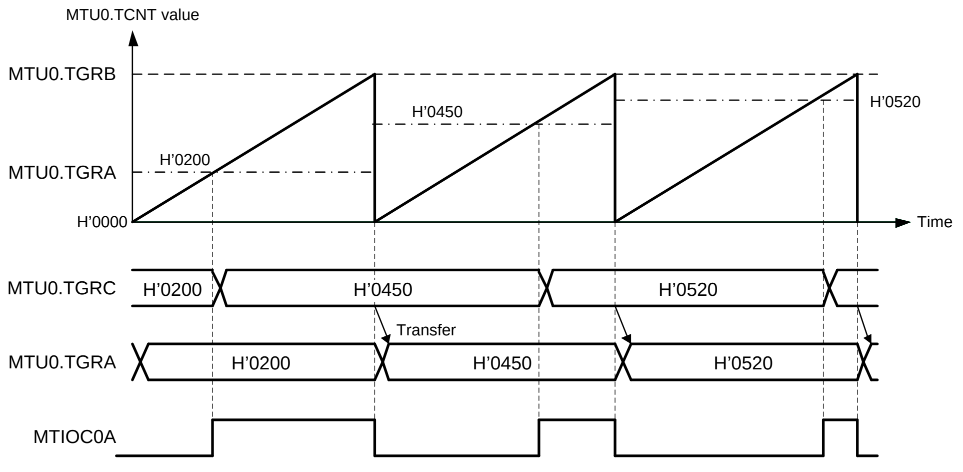


Figure 16.20 Example of Buffer Operation When MTU0.TCNT Clearing is Selected for MTU0.TGRC to MTU0.TGRA Transfer Timing

16.3.4 Cascaded Operation

In cascaded operation, two 16-bit counters in different channels are used together as a 32-bit counter.

There are two functions for connecting MTU1 and MTU2 to use as a 32-bit counter: cascade connection to be set when the MTU1.TMDR3.LWA bit is 0, and cascade connection 32-bit phase counting mode to be set when the MTU1.TMDR3.LWA bit is 1. For details on cascade connection 32-bit phase counting mode, refer to **Section 16.3.6.2, Cascade Connection 32-Bit Phase Counting Mode**. This section describes the cascade connection function to be set when the MTU1.TMDR3.LWA bit is 0.

This function works when the LWA bit of TMDR3 in MTU1 is set to 0 and the TPSC[2:0] bits of TCR in MTU1 are set so that TCNT in MTU1 counts at an overflow/underflow of TCNT in MTU2.

Underflow occurs only when the MTU2 to which the lower 16 bits allocated is in phase counting mode.

Table 16.63 shows the register combinations used in cascaded operation.

NOTE

When phase counting mode is set for MTU1, the count clock setting is invalid and the counters operate independently in phase counting mode.

Table 16.63 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
MTU1 and MTU2	MTU1.TCNT	MTU2.TCNT

For simultaneous input capture of MTU1.TCNT and MTU2.TCNT during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). The input-capture condition is of edges in the signal produced by taking the logical OR of the input level on the main input pin and the input level on the added input pin. Accordingly, if either is at the high level, a change in the level of the other will not produce an edge for detection. For details, see **Section 16.3.4(4), Cascaded Operation Example (c)**. For input capture in cascade connection, refer to **Section 16.6.20, Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection**.

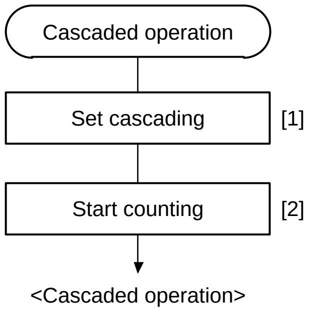
Table 16.64 shows the TICCR setting and input capture input pins.

Table 16.64 TICCR Setting and Input Capture Input Pins

Target Input Capture	TICCR Setting	Input Capture Input Pin
Input capture from MTU1.TCNT to MTU1.TGRA	I2AE bit = 0 (Initial value)	MTIOC1A
	I2AE bit = 1	MTIOC1A, MTIOC2A
Input capture from MTU1.TCNT to MTU1.TGRB	I2BE bit = 0 (Initial value)	MTIOC1B
	I2BE bit = 1	MTIOC1B, MTIOC2B
Input capture from MTU2.TCNT to MTU2.TGRA	I1AE bit = 0 (Initial value)	MTIOC2A
	I1AE bit = 1	MTIOC2A, MTIOC1A
Input capture from MTU2.TCNT to MTU2.TGRB	I1BE bit = 0 (Initial value)	MTIOC2B
	I1BE bit = 1	MTIOC2B, MTIOC1B

(1) Example of Cascaded Operation Setting Procedure

Figure 16.21 shows an example of the cascaded operation setting procedure.



[1] Set the TPSC[2:0] bits in TCR to 111b in MTU1 to select counting at a TCNT overflow/underflow in MTU2.

[2] Set the CSTn bits in TSTRA for the upper and lower channels to 1 to start the count operation.

Figure 16.21 Cascaded Operation Setting Procedure

(2) Cascaded Operation Example (a)

Figure 16.22 shows the operation when MTU1.TCNT is set for counting at MTU2.TCNT overflow/underflow and MTU2 is set for phase counting mode 1 while MTU1.TCNT and MTU2.TCNT are cascaded.

MTU1.TCNT is incremented by MTU2.TCNT overflow and decremented by MTU2.TCNT underflow.

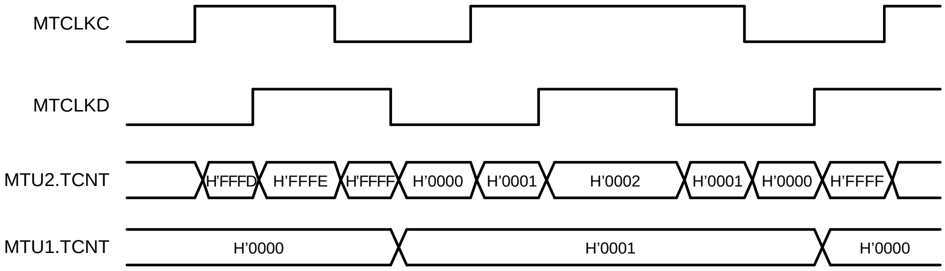


Figure 16.22 Cascaded Operation Example (a)

(3) Cascaded Operation Example (b)

Figure 16.23 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the I2AE bit in TICCR has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the MTU1.TIOR.IOA[3:0] bits have selected the MTIOC1A rising edge for the input capture timing while the MTU2.TIOR.IOA[3:0] bits have selected the MTIOC2A rising edge for the input capture timing.

Under these conditions, the rising edge of both MTIOC1A and MTIOC2A is used for the MTU1.TGRA input capture condition. For the MTU2.TGRA input capture condition, the MTIOC2A rising edge is used.

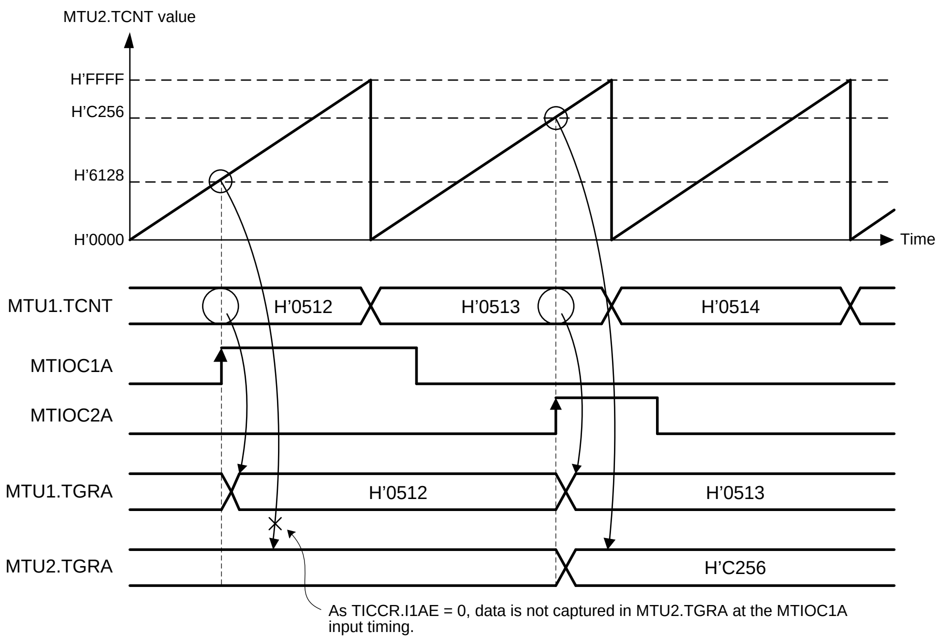
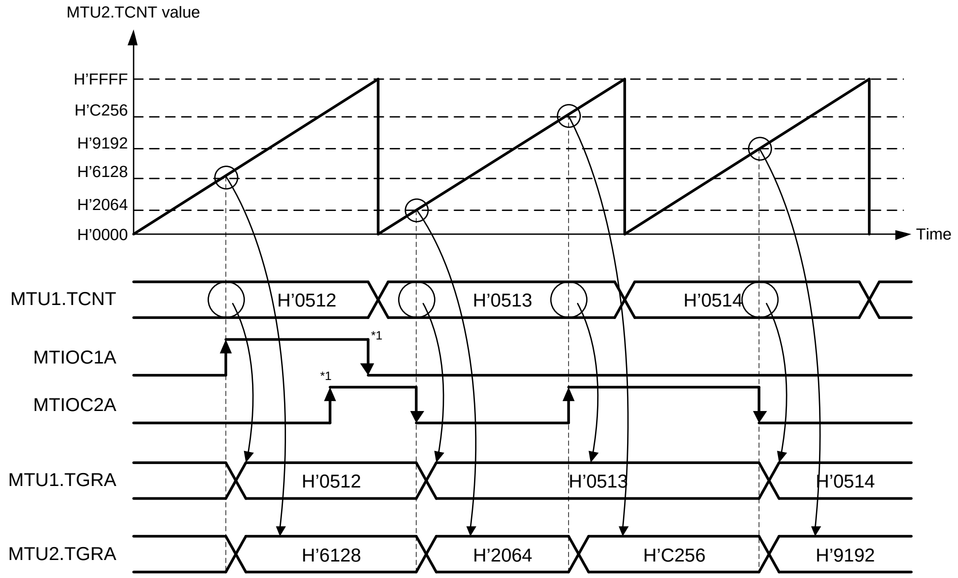


Figure 16.23 Cascaded Operation Example (b)

(4) Cascaded Operation Example (c)

Figure 16.24 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICCR.I2AE and I1AE bits have been set to 1 to include the MTIOC2A and MTIOC1A pins in the MTU1.TGRA and MTU2.TGRA input capture conditions, respectively. In this example, the IOA[3:0] bits in both MTU1.TIOR and MTU2.TIOR have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of MTIOC1A and MTIOC2A input is used for the MTU1.TGRA and MTU2.TGRA input capture conditions.



Note 1. When either of the input signals is driven high, the selected edge of the other cannot act as the input capture condition.

Figure 16.24 Cascaded Operation Example (c)

(5) Cascaded Operation Example (d)

Figure 16.25 illustrates the operation when MTU1.TCNT and MTU2.TCNT have been cascaded and the TICCR.I2AE bit has been set to 1 to include the MTIOC2A pin in the MTU1.TGRA input capture conditions. In this example, the IOA[3:0] bits in MTU1.TIOR have selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing while the IOA[3:0] bits in MTU2.TIOR have selected the MTIOC2A rising edge for the input capture timing.

Under these conditions, as MTU1.TIOR has selected occurrence of MTU0.TGRA compare match or input capture for the input capture timing, the MTIOC2A edge is not used for MTU1.TGRA input capture condition although the I2AE bit in TICCR has been set to 1.

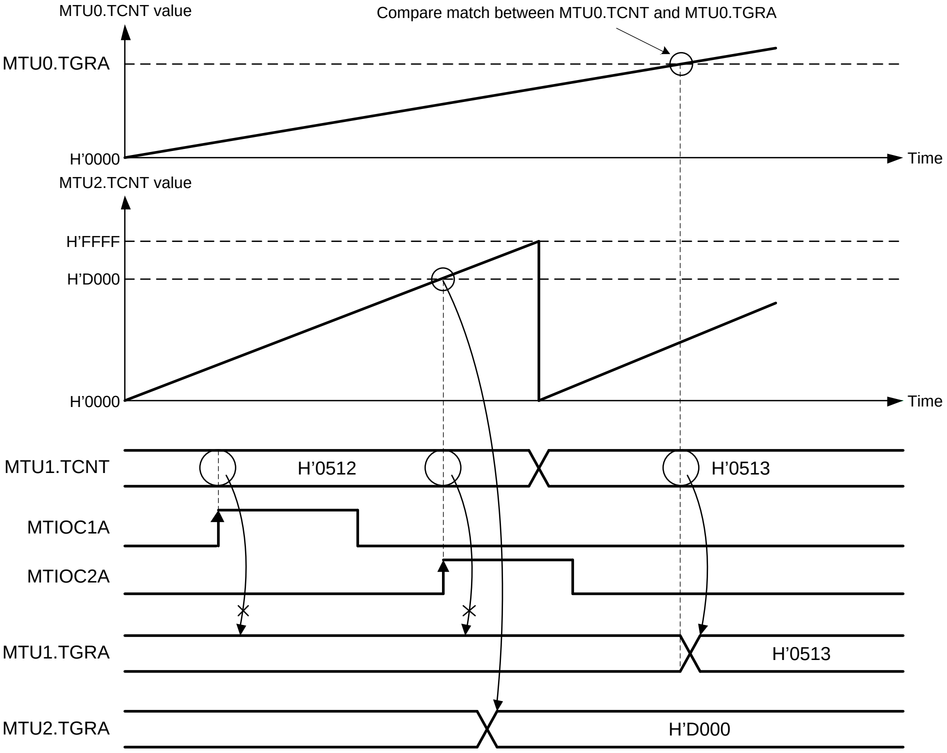


Figure 16.25 Cascaded Operation Example (d)

16.3.5 PWM Modes

PWM modes are provided to output PWM waveforms from the external pins. The output level can be selected as low, high, or toggle output in response to a compare match of each TGR.

PWM waveforms in the range of 0% to 100% duty cycle can be output according to the TGR settings.

By designating TGR compare match as the counter clearing source, the PWM cycle can be specified in that register. Every channel except MTU5 and MTU8 can be set to PWM mode independently. Synchronous operation is also possible between the channels placed in PWM mode or between a channel in PWM mode and another channel in a different mode.

There are two PWM modes as described below.

PWM Mode 1

PWM waveforms are output from the MTIOCnA and MTIOCnC pins by pairing TGRA with TGRB and TGRC with TGRD. The levels specified by the TIOR.IOA[3:0] and IOC[3:0] bits are output from the MTIOCnA and MTIOCnC pins at compare matches A and C, and the level specified by the TIOR.IOB[3:0] and IOD[3:0] bits are output at compare matches B and D (n = 0 to 4, 6, 7). The initial output value is set in TGRA or TGRC. If the values set in paired TGRs are identical, the output value does not change even when a compare match occurs.

In PWM mode 1, PWM waveforms in up to 12 phases can be output.

PWM Mode 2

PWM waveform output is generated using one TGR as the cycle register and the others as duty registers. The level specified in TIOR is output at compare matches. Upon counter clearing by a cycle register compare match, the initial value set in TIOR is output from each pin. If the values set in the cycle and duty registers are identical, the output value does not change even when a compare match occurs.

In PWM mode 2, up to eight phases of PWM waveforms can be output when synchronous clearing is used as synchronous operation in the channels that cannot be placed in PWM mode 2.

The correspondence between PWM output pins and registers is shown in **Table 16.65**.

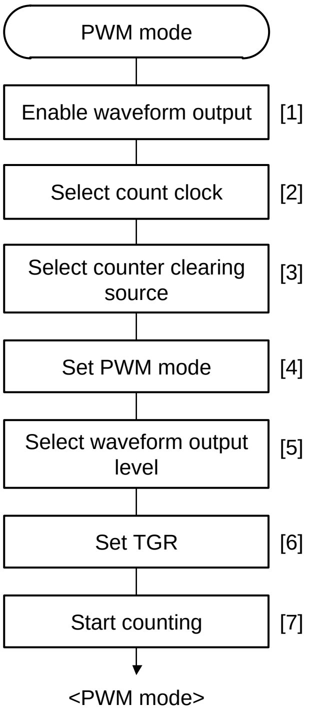
Table 16.65 PWM Output Registers and Output Pins

		Output Pins	
Channel	Register	PWM Mode 1	PWM Mode 2
MTU0	TGRA	MTIOC0A	MTIOC0A
	TGRB		MTIOC0B
	TGRC	MTIOC0C	MTIOC0C
	TGRD		MTIOC0D
MTU1	TGRA	MTIOC1A	MTIOC1A
	TGRB		MTIOC1B
MTU2	TGRA	MTIOC2A	MTIOC2A
	TGRB		MTIOC2B
MTU3	TGRA	MTIOC3A	Setting prohibited
	TGRB		
	TGRC	MTIOC3C	
	TGRD		
MTU4	TGRA	MTIOC4A	
	TGRB		
	TGRC	MTIOC4C	
	TGRD		
MTU6	TGRA	MTIOC6A	
	TGRB		
	TGRC	MTIOC6C	
	TGRD		
MTU7	TGRA	MTIOC7A	
	TGRB		
	TGRC	MTIOC7C	
	TGRD		

Note: In PWM mode 2, PWM waveform output is not possible for the TGR register in which the PWM cycle is set.

(1) Example of PWM Mode Setting Procedure

Figure 16.26 shows an example of the PWM mode setting procedure.



[1] Enable TOERA output when outputting a waveform from the MTIOC pin of MTU3 and MTU4. Enable TOERB output when outputting a waveform from the MTIOC pin of MTU6 and MTU7.

[2] Select the count clock source with bits TPSC[2:0] in TCR. At the same time, select the input edge of the count clock source with bits CKEG[1:0] in TCR.

[3] Use bits CCLR[2:0] in TCR to select the TGR to be used as the TCNT clearing source.

[4] Select the PWM mode with bits MD[3:0] in TMDR.

[5] Use TIOR to designate the TGR as an output compare register, and select the initial value and output value.

[6] Set the cycle in the TGR selected in [3], and set the duty ratio in the other TGR.

[7] Set the CSTn bit in TSTRA or TSTRB to 1 to start the count operation.

Figure 16.26 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 16.27 shows an example of operation in PWM mode 1.

In this example, TGRA compare match is set as the TCNT clearing source, 0 is set as the initial output value and output value for TGRA, and 1 is set as the output value for TGRB.

In this case, the value set in TGRA is used as the cycle, and the value set in TGRB is used as the duty ratio.

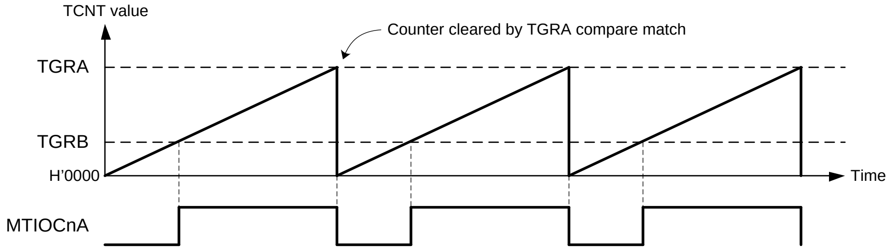


Figure 16.27 Example of PWM Mode 1 Operation (n = 0 to 4, 6, 7)

Figure 16.28 shows an example of operation in PWM mode 2.

In this example, synchronous operation is designated for MTU0 and MTU1, MTU1.TGRB compare match is set as the TCNT clearing source, and low is set as the initial output value and high as the output value for the other TGR registers (MTU0.TGRA to MTU0.TGRD and MTU1.TGRA), outputting 5-phase PWM waveforms.

In this case, the value set in MTU1.TGRB is used as the cycle, and the values set in the other TGRs are used as the duty ratio.

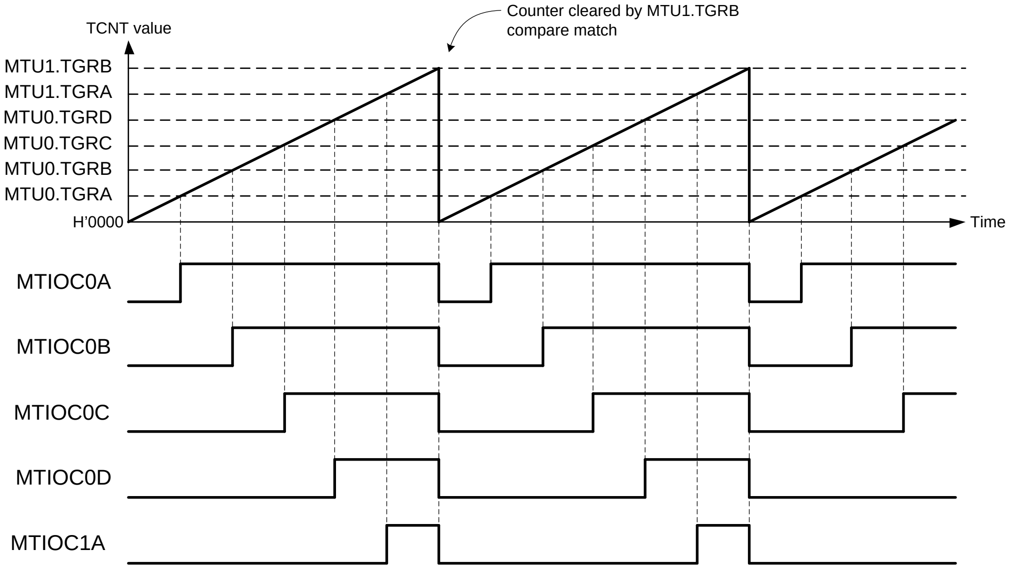


Figure 16.28 Example of PWM Mode 2 Operation

Figure 16.29 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode 1. In these examples, TGRA compare match is selected as the TCNT clearing source, the initial output value and the output value for TGRA are set to the low level, and the output value for TGRB is set to the high level.

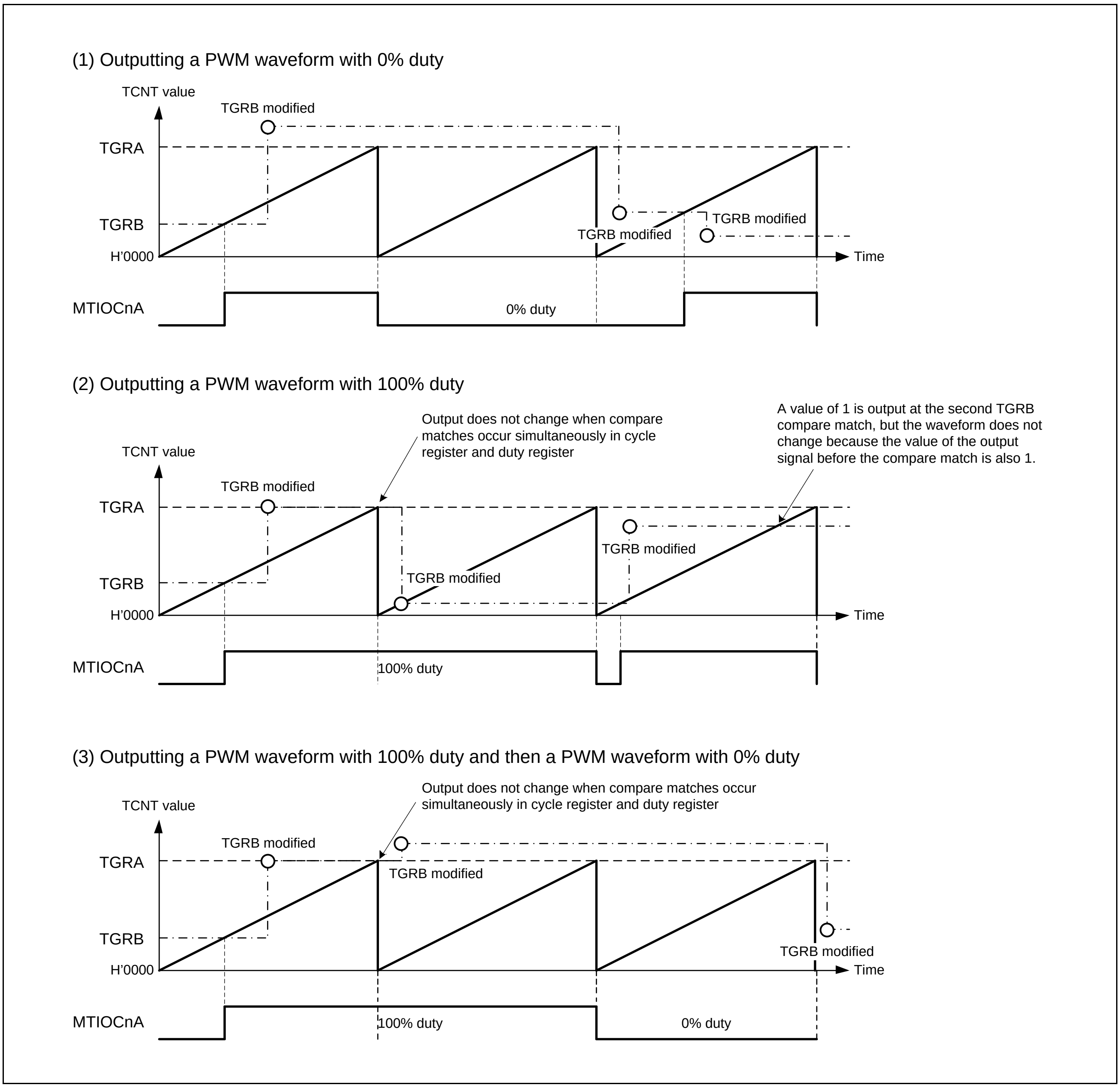


Figure 16.29 Examples of PWM Mode Operation (PWM Waveform Output with 0% Duty and 100% Duty) (n = 0 to 4, 6, 7)

16.3.6 Phase Counting Mode

There are two phase counting modes: 16-bit phase counting mode in which MTU1 and MTU2 operate independently, and cascade connection 32-bit phase counting mode in which MTU1 and MTU2 are cascaded.

In phase counting mode, the phase difference between two external input clocks is detected and the corresponding TCNT is incremented or decremented.

Two external clock input pins for each phase counting mode are not affected by the settings of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. The two external clock input pins used in 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2 can be selected by MTU1.TMDR3.PHCKSEL. In a phase counting mode other than 16-bit phase counting mode and cascade connection 32-bit phase counting mode of MTU2, MTCLKA and MTCLKB are selected for A-phase and B-phase, respectively. In phase counting mode, the external clock pins MTCLKA, MTCLKB, MTCLKC, and MTCLKD are used for two-phase encoder pulse input.

Table 16.66 lists the external clock input pins to be connected in each phase counting mode.

Table 16.66 Clock Input Pins in Phase Counting Mode

Phase Counting Mode	TMDR3.PHCKSEL bit	External Clock Input Pins	
		A-Phase	B-Phase
MTU1 16-bit phase counting mode	x (Don’t care)	MTCLKA	MTCLKB
MTU2 16-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD
Cascade connection 32-bit phase counting mode	0	MTCLKA	MTCLKB
	1 (initial value)	MTCLKC	MTCLKD

16.3.6.1 16-Bit Phase Counting Mode

When the MTU1.TMDR3.LWA is 0, 16-bit phase counting mode can be set individually for MTU1 and MTU2.

In 16-bit phase counting mode, the phase difference between two external input clocks is detected and the 16-bit counter TCNT of the corresponding channel is incremented or decremented.

When 16-bit phase counting mode is specified, an external clock is selected as the count clock and TCNT operates as an up-counter/down-counter regardless of the setting of TCR.TPSC[2:0], TCR2.TPSC2[2:0], and CKEG[1:0]. However, the functions of TCR.CCLR[1:0],TIOR, TIER, and TGR are enabled, and input capture/compare match and interrupt functions can be used.

These external input pins can be used for two-phase encoder pulse input.

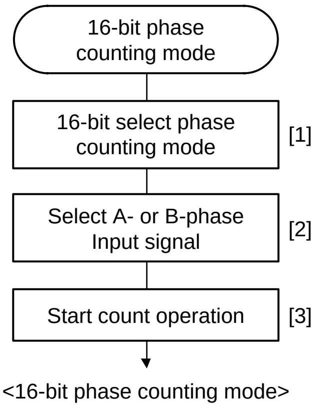
If an overflow occurs during TCNT up-counting, a TCIV interrupt is generated when the corresponding TCIEV bit in the TIER register is set to 1.

If an underflow occurs during TCNT down-counting, a TCIU interrupt is generated when the corresponding TCIEU bit in the TIER register is set to 1.

The TCFD bit in TSR is the count direction flag. Read the TCFD flag to check whether TCNT is counting up or down.

(1) Example of 16-Bit Phase Counting Mode Setting Procedure

Figure 16.30 shows an example of the phase counting mode setting procedure.



[1] Set the TMDR3.LWA bit of MTU1 to 0. Set the TMDR1.MD[3:0] bits to select the phase counting mode.

[2] When the MTU2 is set to phase counting mode, select the A- or B-phase input signal with the PHCKSEL bit in TMDR3. This setting is not necessary for MTU1 because MTCLKA and MTCLKB are always selected for A-phase and B-phase, respectively.

[3] Set the CSTn bit in TSTRA to 1 to start the count operation.

Figure 16.30 Example of 16-Bit Phase Counting Mode Setting Procedure

(2) Examples of 16-Bit Phase Counting Mode Operation

In phase counting mode, TCNT is incremented or decremented according to the phase difference between two external clocks. There are five modes according to the count conditions. Each mode operates under the condition PHCKSEL = 1, which means the phase clock for MTU1 is input from MTCLKA or MTCLKB and that for MTU2 is input from MTCLKC or MTCLKD.

(a) Phase Counting Mode 1

Figure 16.31 shows an example of operation in phase counting mode 1, and **Table 16.67** summarizes the TCNT up- counting and down-counting conditions.

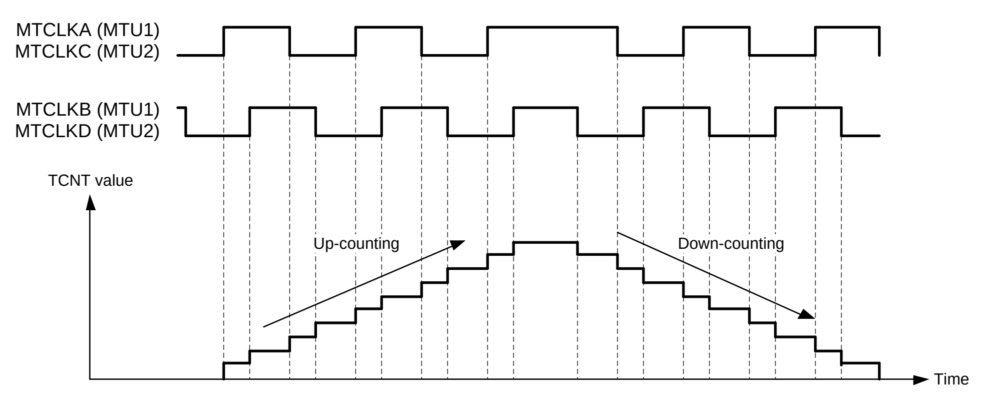


Figure 16.31 Example of Operation in Phase Counting Mode 1

Table 16.67 Up-Counting and Down-Counting Conditions in Phase Counting Mode 1

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High	↑	Up-counting
Low	↓	
↑	Low	
↓	High	
High	↓	Down-counting
Low	↑	
↑	High	
↓	Low	

Remarks: ↑: Rising edge
↓: Falling edge

(b) Phase Counting Mode 2

Figure 16.32 to **Figure 16.34** show the examples of operation in phase counting mode 2 and **Table 16.68** summarizes the TCNT up-counting and down-counting conditions.

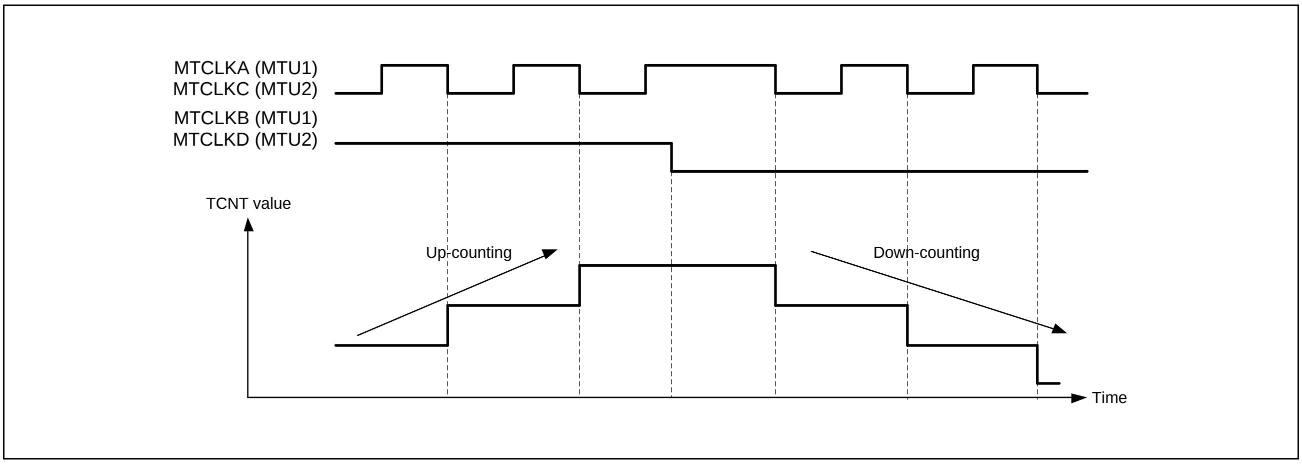


Figure 16.32 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2))

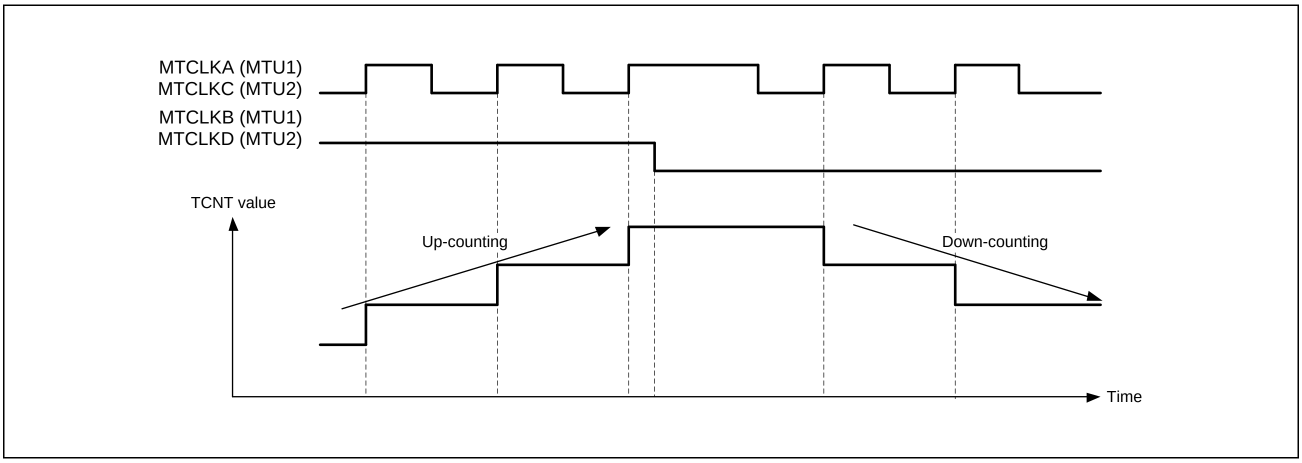


Figure 16.33 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2))

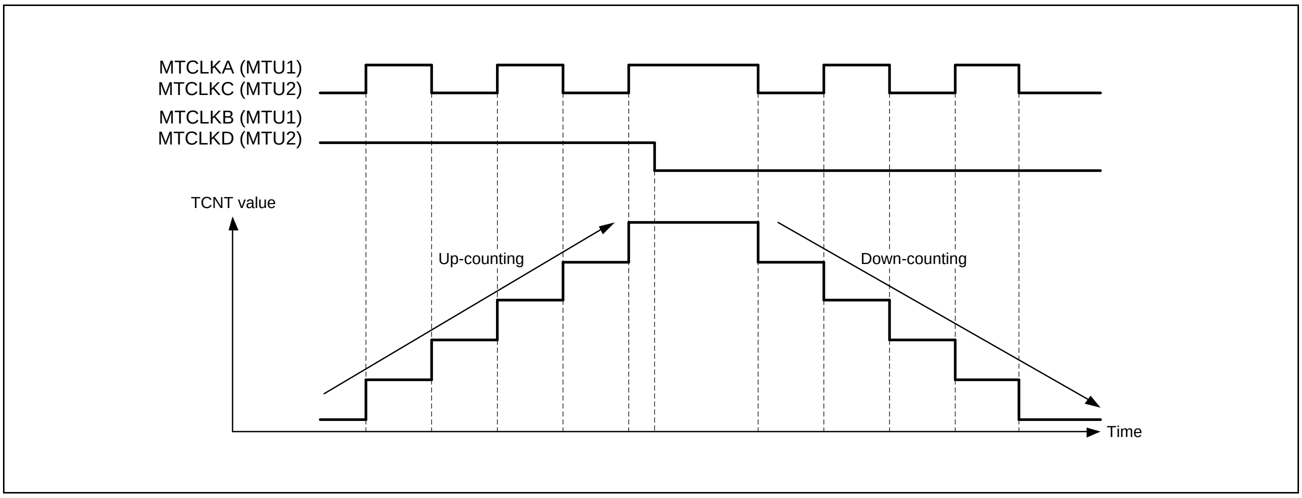


Figure 16.34 Example of Operation in Phase Counting Mode 2 (When MTUn.TCR2.PCB[1:0] is 1xb (n = 1, 2))

Table 16.68 Up-Counting and Down-Counting Conditions in Phase Counting Mode 2

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High	↑	Not counted (Don’t care)
	Low	↓	
	↑	Low	
	↓	High	Up-counting
	High	↓	
	Low	↑	
	↑	High	
	↓	Low	Down-counting
01b	High	↑	Not counted (Don’t care)
	Low	↓	
	↑	Low	Down-counting
	↓	High	Not counted (Don’t care)
	High	↓	
	Low	↑	
	↑	High	Up-counting
	↓	Low	Not counted (Don’t care)
1xb	High	↑	Not counted (Don’t care)
	Low	↓	
	↑	Low	Down-counting
	↓	High	Up-counting
	High	↓	Not counted (Don’t care)
	Low	↑	
	↑	High	Up-counting
	↓	Low	Down-counting

Remarks: ↑ : Rising edge
↓ : Falling edge

(c) Phase Counting Mode 3

Figure 16.35 to **Figure 16.37** show the examples of operation in phase counting mode 3 and **Table 16.69** summarizes the TCNT up-counting and down-counting conditions.

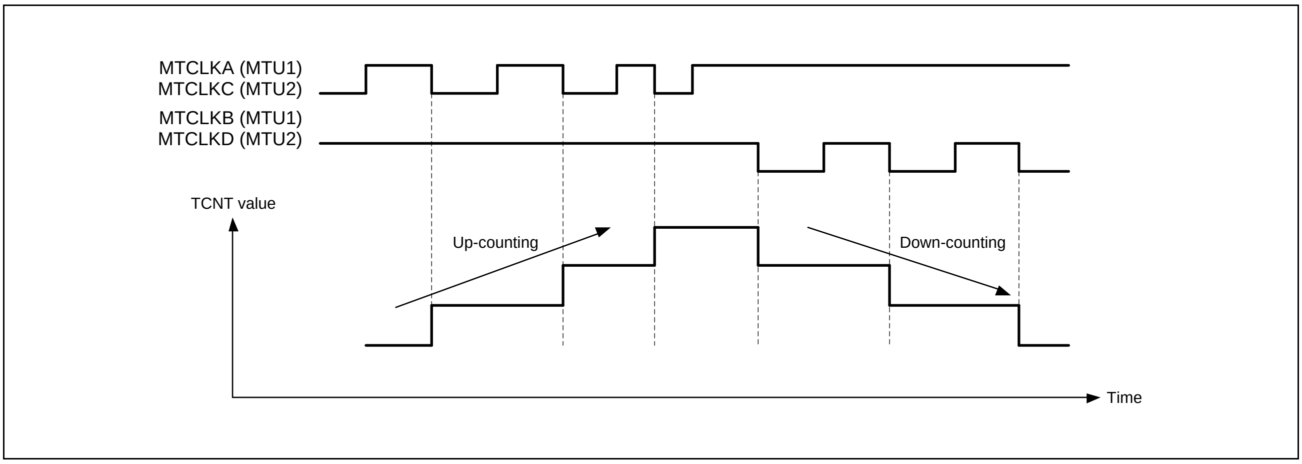


Figure 16.35 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 00b (n = 1, 2))

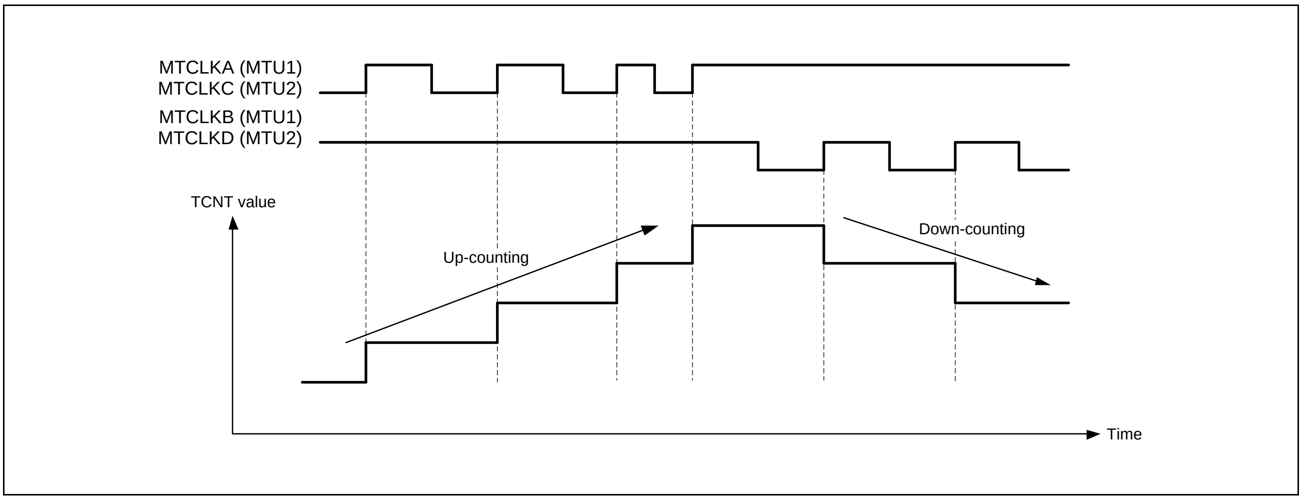


Figure 16.36 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 01b (n = 1, 2))

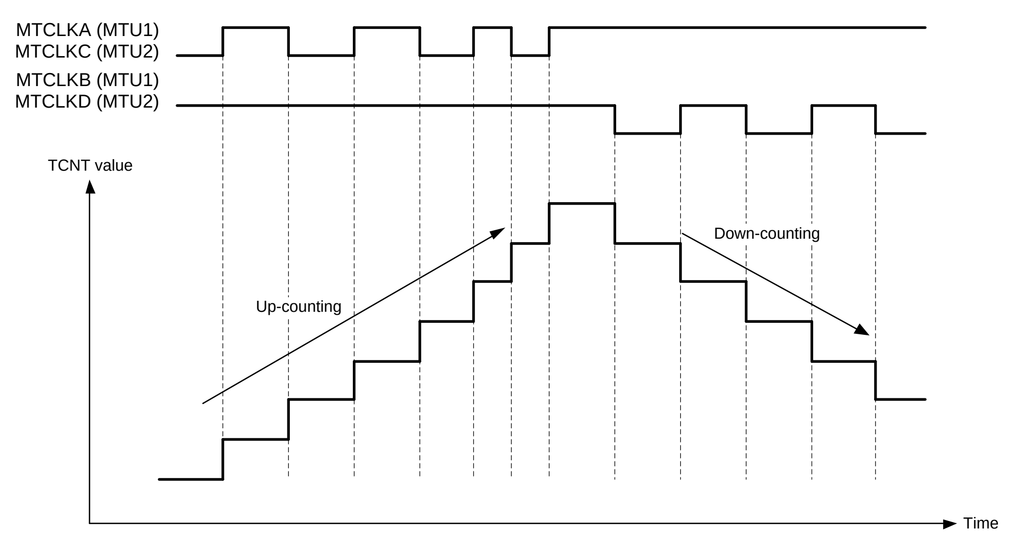


Figure 16.37 Example of Operation in Phase Counting Mode 3 (When MTUn.TCR2.PCB[1:0] is 1xb (n = 1, 2))

Table 16.69 Up-Counting and Down-Counting Conditions in Phase Counting Mode 3

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
00b	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	
	↓	High	Up-counting
	High	↓	Down-counting
	Low	↑	Not counted (Don't care)
	↑	High	
	↓	Low	
01b	High	↑	Down-counting
	Low	↓	Not counted (Don't care)
	↑	Low	
	↓	High	
	High	↓	
	Low	↑	
	↑	High	Up-counting
	↓	Low	Not counted (Don't care)
1xb	High	↑	Down-counting
	Low	↓	Not counted (Don't care)
	↑	Low	
	↓	High	Up-counting
	High	↓	Down-counting
	Low	↑	Not counted (Don't care)
	↑	High	Up-counting
	↓	Low	Not counted (Don't care)

Remarks: ↑: Rising edge
↓: Falling edge

(d) Phase Counting Mode 4

Figure 16.38 shows an example of operation in phase counting mode 4, and **Table 16.70** summarizes the TCNT up- counting and down-counting conditions.

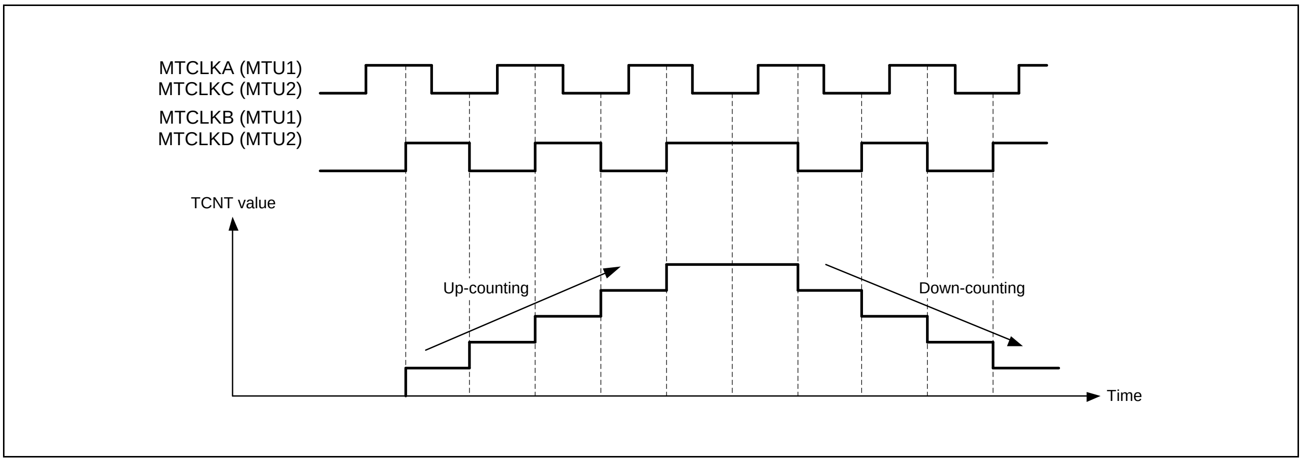


Figure 16.38 Example of Operation in Phase Counting Mode 4

Table 16.70 Up-Counting and Down-Counting Conditions in Phase Counting Mode 4

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
High	↑	Up-counting
Low	↓	
↑	Low	Not counted (Don’t care)
↓	High	
High	↓	Down-counting
Low	↑	
↑	High	Not counted (Don’t care)
↓	Low	

Remarks: ↑: Rising edge
↓: Falling edge

(e) Phase Counting Mode 5

Figure 16.39 and **Figure 16.40** show the examples of operation in phase counting mode 5 and **Table 16.71** summarizes the TCNT up-counting and down-counting conditions.

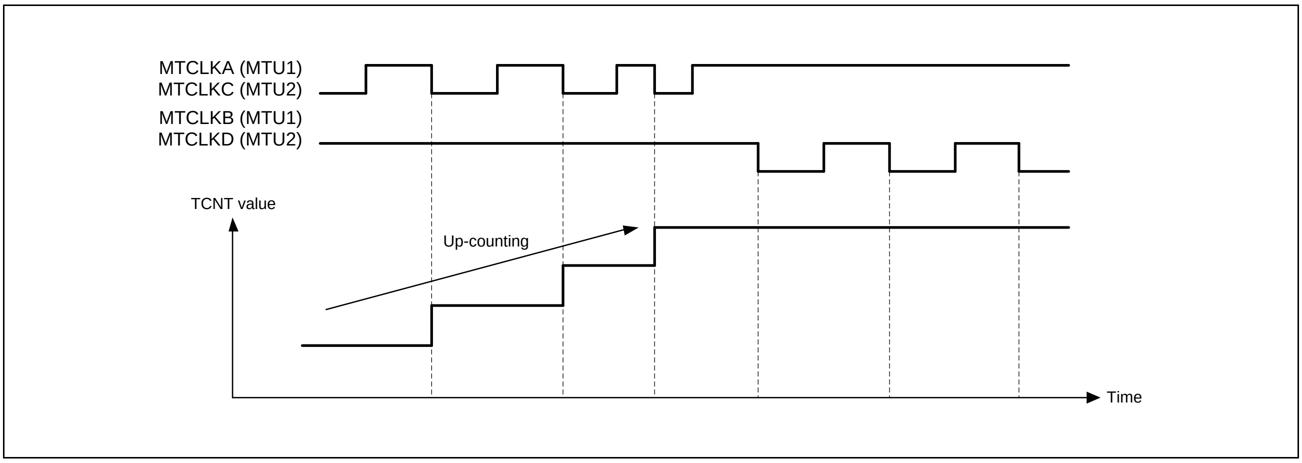


Figure 16.39 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 0xb (n = 1, 2))

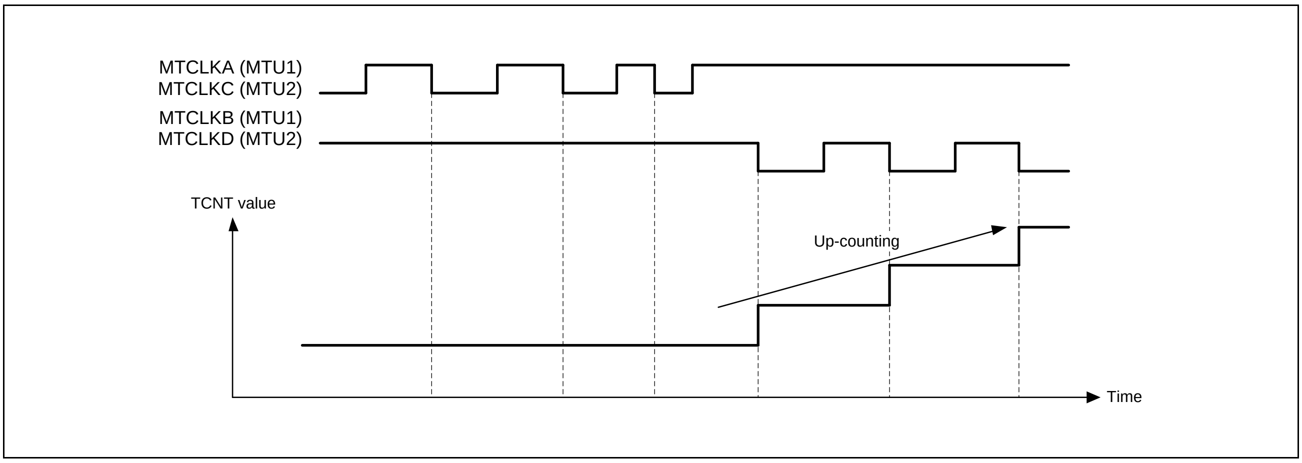


Figure 16.40 Example of Operation in Phase Counting Mode 5 (When MTUn.TCR2.PCB[1:0] = 1xb (n = 1, 2))

Table 16.71 Up-Counting and Down-Counting Conditions in Phase Counting Mode 5

PCB[1:0]	MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	Operation
0xb	High	↑	Not counted (Don't care)
	Low	↓	
	↑	Low	
	↓	High	Up-counting
	High	↓	Not counted (Don't care)
	Low	↑	
	↑	High	
	↓	Low	Up-counting
1xb	High	↑	Not counted (Don't care)
	Low	↓	Up-counting
	↑	Low	Not counted (Don't care)
	↓	High	
	High	↓	Up-counting
	Low	↑	Not counted (Don't care)
	↑	High	
	↓	Low	

Remarks: ↑: Rising edge
↓: Falling edge

(3) 16-Bit Phase Counting Mode Application Example

Figure 16.41 shows an example in which MTU1 is in phase counting mode, and MTU1 is coupled with MTU0 to input 2-phase encoder pulses of a servo motor in order to detect position or speed.

MTU1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to MTCLKA and MTCLKB.

In MTU0, MTU0.TGRC compare match is specified as the TCNT clearing source and MTU0.TGRA and MTU0.TGRC are used for the compare match function and are set with the speed control cycle and position control cycle.

MTU0.TGRB is used for input capture, with MTU0.TGRB and MTU0.TGRD operating in buffer mode. The MTU1 count clock is designated as the MTU0.TGRB input capture source, and the widths of 2-phase encoder 4-multiplication pulses are detected.

MTU1.TGRA and MTU1.TGRB for MTU1 are designated for the input capture function and MTU0.TGRA and MTU0.TGRC compare matches in MTU0 are selected as the input capture sources to store the up/down-counter values for the control cycles.

This procedure enables the accurate detection of position and speed.

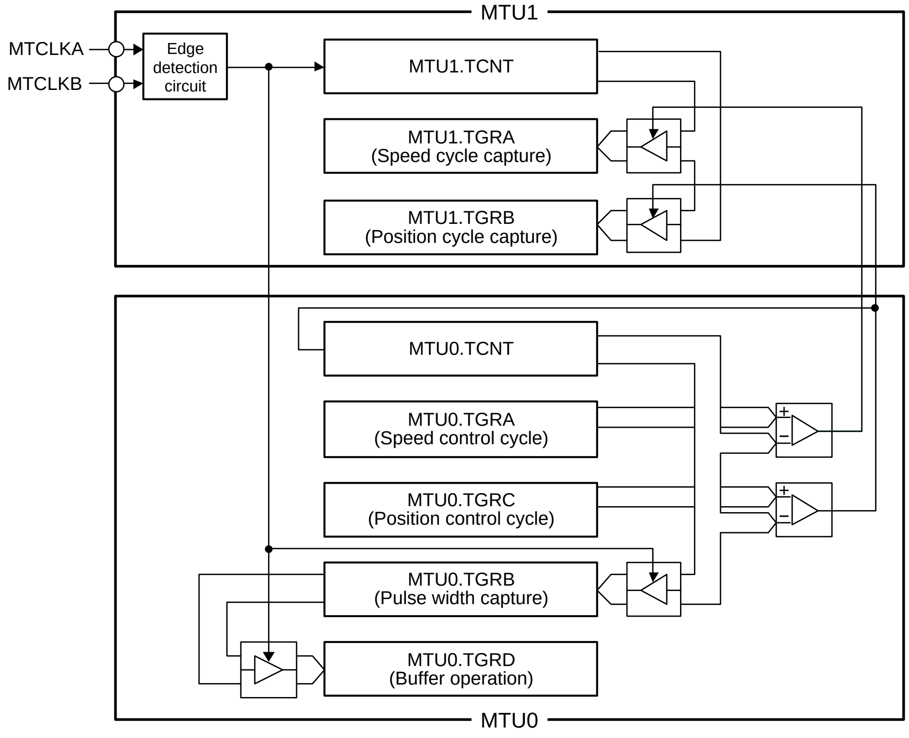


Figure 16.41 16-Bit Phase Counting Mode Application Example

16.3.6.2 Cascade Connection 32-Bit Phase Counting Mode

When MTU1 is set to phase counting mode by setting MTU1.TMDR3.LWA = 1, MTU1 and MTU2 are connected to operate in cascade connection 32-bit phase counting mode as shown in **Figure 16.42**. When this mode is used, the TCR, TCR2, TIOR, TIER, TGR, and TSR registers are controlled by MTU1 and the settings of MTU2 are disabled. See **Figure 16.43** for the procedure for setting cascade connection 32-bit phase counting mode.

In this mode, three-phase (A, B, and Z) signals can be input. As an encoder pulse signal, the external input phase clocks MTCLKA and MTCLKB or MTCLKC and MTCLKD can be selected for A-phase and B-phase, and MTIOC1A can be selected for Z-phase, respectively. See **Table 16.70** for selecting external clock input of A-phase and B-phase. A counter event is generated using an A-phase or B-phase pulse and counted by the 32-bit counter MTU1.TCNTLW.

An input capture can also be generated using a Z-phase signal; thus, an angular velocity can be measured using the value captured in the general register.

Furthermore, MTU8 can be used as a channel for measuring a 1-ms interval, and a compare match signal can be output at a 1-ms interval to the MTU1 and MTU2, which operate in cascade connection 32-bit phase counting mode. That is, a compare match signal of MTU8 is used as a capture signal of MTU1 and MTU2, and the number of A-phase and B-phase pulses for a 1-ms period can be measured.

When MTU0 or MTU5 is specified as the channel for measuring Z-phase signal pulses, this compare match signal of TGRC in MTU8 can be output as a capture signal or a clear signal to MTU0 or MTU5; thus, the Z-phase count can be measured at 1-ms intervals.

In addition, a counter event signal of combined MTU1 and MTU2 can be used as a capture signal of TGRD in MTU8, and measurement can be performed including the intervals of A, B, or both phase pulses. In this case, the TGRD register in MTU8 should be set to buffer operation.

Refer to **Section 16.3.4, Cascaded Operation**, for details on the cascade connection function for connecting MTU1 and MTU2 in a mode other than cascade connection 32-bit phase counting mode.

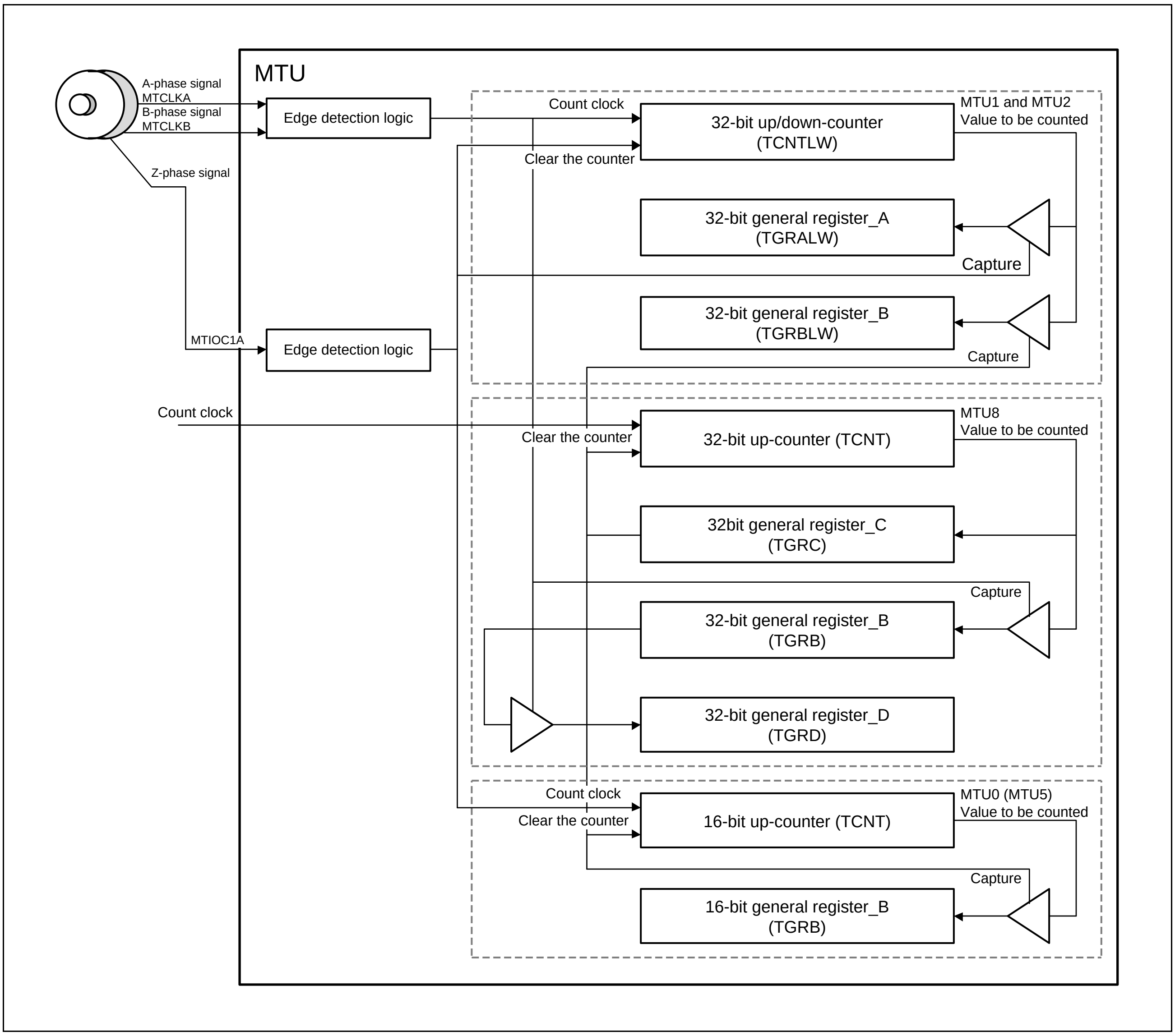
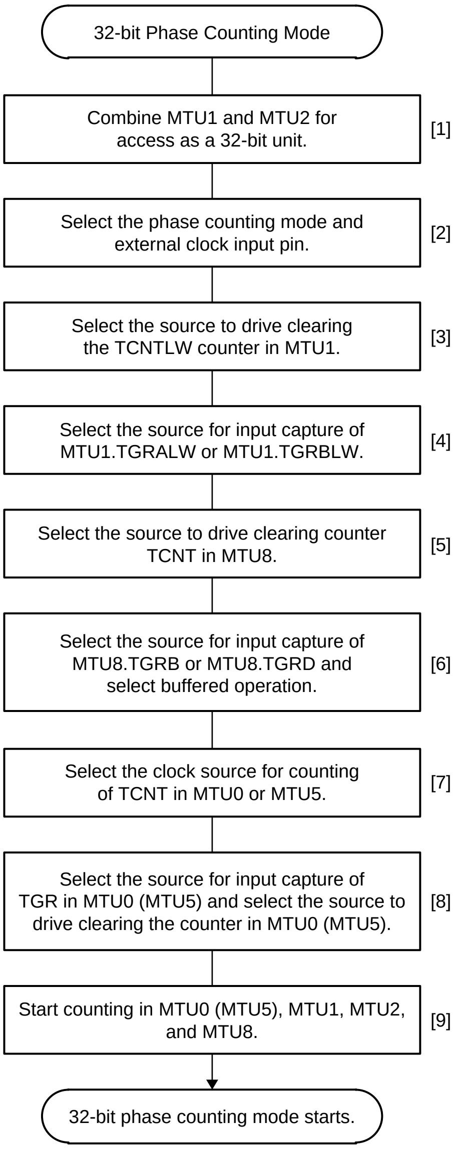


Figure 16.42 Block Diagram for Operation in Cascade Connection 32-Bit Phase Counting Mode

(1) Example of Setting Cascade Connection 32-Bit Phase Counting Mode

Figure 16.43 shows an example of the procedure for setting cascade connection 32-bit phase counting mode.



[1] Set the LWA bit in MTU1.TMDR3 to 1 and combine MTU1 and MTU3 for access as a 32-bit unit.*1

[2] Select phase counting mode with the MD[3:0] bits in MTU1.TMDR1 and select the external clock input pin with the PHCKSEL bits in MTU1.TMDR3.

[3] Set the CCLR[2:0] bits of TCR in MTU1 to 001b to select input capture of TGRALW in MTU1 as the source of the signal to drive clearing the TCNTLW counter in MTU1.

[4] Set the IOA[3:0] bits in MTU1.TIOR to select the effective edge (rising, falling or both) of the signal produced by input capture of TGRALW in MTU1. After that, set the IOB[3:0] bits in MTU1.TIOR to 1110b to select the compare match signal of MTU8.TGRC as the source to drive input capture of MTU1.TGRBLW.

[5] Set the CCLR[2:0] bits in MTU8.TCR to 101b to select the compare match signal of MTU8.TGRC as the source to drive clearing counter TCNT in MTU8.

[6] Select the source for input capture of MTU8.TGRB as the clock for counting of TCNTLW in MTU1 by setting the IOB[3:0] bits in MTU8.TIORH to 1110b. At the same time, select buffered operation for TGRB and TGRD by setting the BFB bit in MTU8.TMDR1 to 1.

[7] Set the TPSC2[2:0] bits in MTU0.TCR2 to 111b to select the MTIOC1A pin as the clock source for counting of TCNT in MTU0.
(When using TCNT in MTU5, set the TPSC2[2:0] bits in MTU5.TCR2 to 111b).

[8] Select the compare match signal of MTU8.TGRC as the source to drive input capture of MTU0.TGRA by setting the IOA[3:0] bits in MTU0.TIORH to 1110b. (When using TCNT in MTU5, set the TIORU, TIORV, or TIORW bit in MTU5 is set to 10100b).
After that, select input capture of MTU0.TGRA as the source to drive clearing counter TCNT in MTU0 by setting the CCLR[2:0] bits in MTU0.TCR to 001b.

[9] Start counting in MTU0, MTU1, MTU2, and MTU8 by setting the CST0, CST1, CST2, and CST8 bits in TSTRA to 1 at the same time.
(When using TCNT in MTU5, set the counter corresponding to TSTR in MTU5 to 1).

Note 1. When the LWA bit in MTU1.TMDR3 is set to 1, the TCR, TCR2, TMDR, and TIOR registers in MTU1 are effective in controlling cascaded operation. In this case, the settings of these registers in MTU2 are ignored.

Figure 16.43 Procedure for Setting Cascade Connection 32-Bit Phase Counting Mode

16.3.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three phases of positive and negative PWM waveforms (six phases in total) that share a common wave transition point can be output by combining MTU3 and MTU4 and MTU6 and MTU7.

When set for reset-synchronized PWM mode, the MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7C, MTIOC7B, and MTIOC7D pins function as PWM output pins and timer counters 6 and 12 (MTU3.TCNT and MTU6.TCNT) functions as an up-counter.

Table 16.72 shows the PWM output pins used. **Table 16.73** shows the settings of the registers.

Table 16.72 Output Pins for Reset-Synchronized PWM Mode

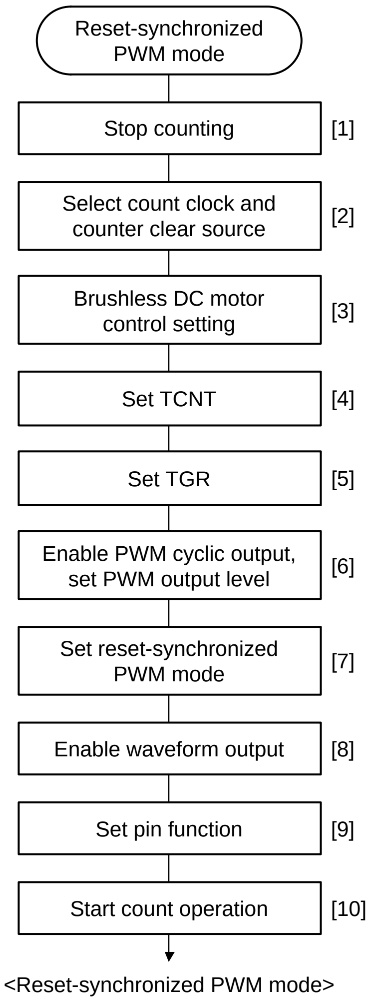
Channel	Output Pin	Description
MTU3	MTIOC3B	PWM output pin 1
	MTIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)
MTU6	MTIOC6B	PWM output pin 4
	MTIOC6D	PWM output pin 4' (negative-phase waveform of PWM output 4)
MTU7	MTIOC7A	PWM output pin 5
	MTIOC7C	PWM output pin 5' (negative-phase waveform of PWM output 5)
	MTIOC7B	PWM output pin 6
	MTIOC7D	PWM output pin 6' (negative-phase waveform of PWM output 6)

Table 16.73 Register Settings for Reset-Synchronized PWM Mode

Register	Setting
MTU3.TCNT	Initial setting (H’0000)
MTU4.TCNT	Initial setting (H’0000)
MTU3.TGRA	Set the count cycle for MTU3.TCNT
MTU3.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC3B and MTIOC3D pins
MTU4.TGRA	Set the transition point of the PWM waveform to be output from the MTIOC4A and MTIOC4C pins
MTU4.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC4B and MTIOC4D pins
MTU6.TCNT	Initial setting (H’0000)
MTU7.TCNT	Initial setting (H’0000)
MTU6.TGRA	Set the count cycle for MTU6.TCNT
MTU6.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC6B and MTIOC6D pins
MTU7.TGRA	Set the transition point of the PWM waveform to be output from the MTIOC7A and MTIOC7C pins
MTU7.TGRB	Set the transition point of the PWM waveform to be output from the MTIOC7B and MTIOC7D pins

(1) Example of Procedure for Setting Reset-Synchronized PWM Mode

Figure 16.44 shows an example of the procedure for specifying the reset-synchronized PWM mode.



[1] Clear the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 0 to stop the TCNT count operation. Specify the reset-synchronized PWM mode while MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) are stopped.

[2] Set the TCR.TPSC[2:0] and TCR.CKEG[1:0] bits of MTU3 (MTU6) to select the count clock source and clock edge for MTU3 (MTU6). Set the TCR.CCLR[2:0] bits to select TGRA compare match as the counter clear source.

[3] To control a brushless DC motor, set the TGCRA.BDC bit and set the feedback signal input source and output chopping or gate signal direct output (only when MTU3 and MTU4 are used).

[4] Set MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) to H'0000.

[5] MTU3.TGRA (MTU6.TGRA) is the cycle register. Set the PWM cycle in MTU3.TGRA (MTU6.TGRA).
Set the PWM output transition timing in MTU3.TGRB (MTU6.TGRB), MTU4.TGRA (MTU7.TGRA), and MTU4.TGRB (MTU7.TGRB).
Note that the setting values should be within the range of compare match with MTU3.TCNT (MTU6.TCNT).

X ≤ MTU3.TGRA (X: Set value)

[6] Set the PSYE bit in the TOCR1A (TOCR1B) register to enable or disable toggle output synchronized with the PMW cycle, and set the OLSP and OLSN bits to select the PWM output level.
When specifying the PWM output level using the TOCR2A (TOCR2B) and TOLBRA (TOLBRB) in buffer operation, see Timer Output Level Buffer Registers (TOLBRA and TOLBRB).

[7] Set the MD[3:0] bits in MTU3.TMDR1 (MTU6.TMDR1) register to 1000b to select the reset-synchronized PWM mode. Specify the buffer operation (BFA = 1) of the MTU3.TGRA (MTU6.TGRA) register and the buffer operation (BFB = 1) of the MTU3.TGRB register when necessary. Set the MTU4.TMDR1 (MTU7.TMDR1) register to the initial value (normal operation; BFA = 0 and BFB = 0).

[8] Set the TOERA (TOERB) register to enable or disable output from the PWM output pin. Then, set the MTU3.TIORH (MTU6.TIORH), MTU3.TIORL (MTU6.TIORL), MTU4.TIORH (MTU7.TIORH), and MTU4.TIORL (MTU7.TIORL) registers to H'00.

[9] Set the port mode control registers (PMCn) and Pmn port fuction control reigister (PFCm) of the GPIO ports.

[10] Set the TSTRA.CST3 (TSTRB.CST6) bit to 1 to start count operation.

Note: The output waveform starts to toggle at the point of MTU3.TCNT (MTU6.TCNT) = MTU3.TGRA (MTU6.TGRA) = X by setting X = MTU3.TGRA (MTU6.TGRA), i.e., cycle = duty ratio.

Figure 16.44 Procedure for Selecting Reset-Synchronized PWM Mode

(2) Example of Reset-Synchronized PWM Mode Operation

Figure 16.45 shows an example of operation in the reset-synchronized PWM mode.

MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) operate as up-counters. The counters are cleared when a compare match occurs between MTU3.TCNT (MTU6.TCNT) and MTU3.TGRA (MTU6.TGRA), and then begin incrementing from H’0000. The output from the PWM pins toggles every time a compare match occurs in MTU3.TGRB (MTU6.TGRB), MTU4.TGRA (MTU7.TGRA), and MTU4.TGRB (MTU7.TGRB) and the counters are cleared.

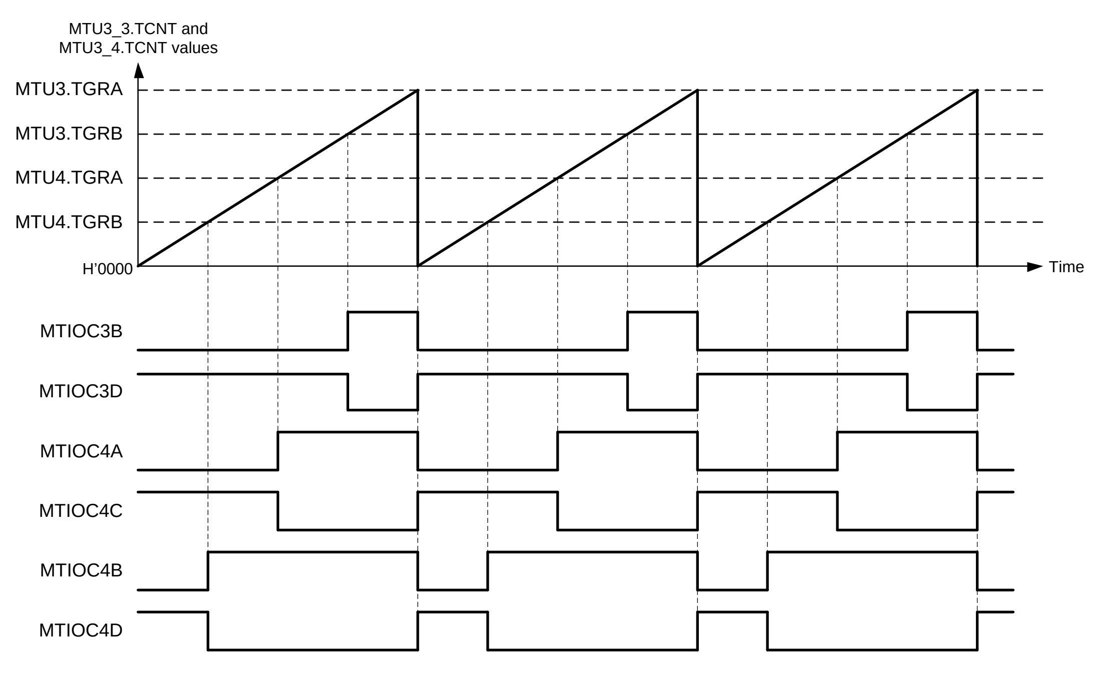


Figure 16.45 Example of Reset-Synchronized PWM Mode Operation
(When OLSN = 1 and OLSP = 1 in MTU3.TOCR1 and MTU4.TOCR1)

16.3.8 Complementary PWM Mode

In complementary PWM mode, dead time can be set for PWM waveforms to be output. The dead time is the period during which the upper and lower arm transistors are set to the inactive level in order to prevent short-circuiting of the arms.

Six positive-phase and six negative-phase PWM waveforms (12 phases in total) with dead time can be output by combining MTU3/ MTU4 and MTU6/MTU7. PWM waveforms without dead time can also be output.

In complementary PWM mode, MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D, MTIOC6B,

MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D pins function as PWM output pins, and the MTIOC3A and MTIOC6A pins can be set for toggle output synchronized with the PWM cycle.

MTU3.TCNT, MTU4.TCNT, MTU6.TCNT, and MTU7.TCNT function as up/down-counters.

Table 16.74 shows the PWM output pins used. **Table 16.75** shows the settings of the registers used.

A function to directly cut off the PWM output by using an external signal is supported as a port function.

Table 16.74 Output Pins for Complementary PWM Mode

Channel	Output Pin	Description
MTU3	MTIOC3A	Toggle output synchronized with PWM cycle (or I/O port)
	MTIOC3B	PWM output pin 1
	MTIOC3C	I/O port*1
	MTIOC3D	PWM output pin 1' (negative-phase waveform output of PWM output 1)
MTU4	MTIOC4A	PWM output pin 2
	MTIOC4C	PWM output pin 2' (negative-phase waveform output of PWM output 1)
	MTIOC4B	PWM output pin 3
	MTIOC4D	PWM output pin 3' (negative-phase waveform output of PWM output 1)
MTU6	MTIOC6A	Toggle output synchronized with PWM cycle (or I/O port)
	MTIOC6B	PWM output pin 4
	MTIOC6C	I/O port*1
	MTIOC6D	PWM output pin 4' (negative-phase waveform output of PWM output 4)
MTU7	MTIOC7A	PWM output pin 5
	MTIOC7C	PWM output pin 5' (negative-phase waveform output of PWM output 5)
	MTIOC7B	PWM output pin 6
	MTIOC7D	PWM output pin 6' (negative-phase waveform output of PWM output 6)

Note 1. Avoid setting the MTIOC3C and MTIOC6C pins as timer I/O pins in complementary PWM mode.

Table 16.75 Register Settings for Complementary PWM Mode (1/2)

Channel	Counter / Register	Description	Read/Write from CPU
MTU3	TCNT	Starts up-counting from the value set in the dead time register	Maskable by TRWERA setting*[1]
	TGRA	Set MTU3.TCNT upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWERA setting*[1]
	TGRB	PWM output 1 compare register	Maskable by TRWERA setting*[1]
	TGRC	MTU3.TGRA buffer register	Readable/writable
	TGRD	PWM output 1/MTU3.TGRB buffer register	Readable/writable
	TGRE	MTU3.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU4	TCNT	Starts up-counting after being initialized to H'0000	Maskable by TRWERA setting*[1]
	TGRA	PWM output 2 compare register	Maskable by TRWERA setting*[1]
	TGRB	PWM output 3 compare register	Maskable by TRWERA setting*[1]
	TGRC	PWM output 2/MTU4.TGRA buffer register	Readable/writable
	TGRD	PWM output 3/MTU4.TGRB buffer register	Readable/writable
	TGRE	MTU4.TGRA buffer register B (when double buffer function is used)	Readable/writable
	TGRF	MTU4.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU6	TCNT	Starts up-counting from the value set in the dead time register	Maskable by TRWERB setting*[2]
	TGRA	Set MTU6.TCNT upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWERB setting*[2]
	TGRB	PWM output 4 compare register	Maskable by TRWERB setting*[2]
	TGRC	MTU6.TGRA buffer register	Readable/writable
	TGRD	PWM output 4/MTU6.TGRB buffer register	Readable/writable
	TGRE	MTU6.TGRB buffer register B (when double buffer function is used)	Readable/writable
MTU7	TCNT	Starts up-counting after being initialized to H'0000	Maskable by TRWERB setting*[2]
	TGRA	PWM output 5 compare register	Maskable by TRWERB setting*[2]
	TGRB	PWM output 6 compare register	Maskable by TRWERB setting*[2]
	TGRC	PWM output 5/MTU7.TGRA buffer register	Readable/writable
	TGRD	PWM output 6/MTU7.TGRB buffer register	Readable/writable
	TGRE	MTU7.TGRA buffer register B (when double buffer function is used)	Readable/writable
	TGRF	MTU7.TGRB buffer register B (when double buffer function is used)	Readable/writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).

Note 2. Access can be enabled or disabled according to the setting in TRWERB (timer read/write enable register B).

Table 16.76 Register Settings for Complementary PWM Mode (2/2)

Counter / Register	Description	Read/Write from CPU
Timer dead time data register A (TDDRA)	Set MTU4.TCNT and MTU3.TCNT offset value (dead time value)	Maskable by TRWERA setting*[1]
Timer dead time data register B (TDDRB)	Set MTU7.TCNT and MTU6.TCNT offset value (dead time value)	Maskable by TRWERB setting*[2]
Timer cycle data register A (TCDRA)	Set MTU4.TCNT upper limit value (1/2 carrier cycle)	Maskable by TRWERA setting*[1]
Timer cycle data register B (TCDRB)	Set MTU7.TCNT upper limit value (1/2 carrier cycle)	Maskable by TRWERB setting*[2]
Timer cycle buffer register A (TCBRA)	TCDRA buffer register	Readable/writable
Timer cycle buffer register B (TCBRB)	TCDRB buffer register	Readable/writable
Subcounter A (TCNTSA)	Subcounter A for dead time generation	Read-only
Subcounter B (TCNTSB)	Subcounter B for dead time generation	Read-only
Temporary register 1A (TEMP1A)	PWM output 1/MTU3.TGRB temporary register A	Not readable or writable
Temporary register 1B (TEMP1B)	PWM output 1/MTU3.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 2A (TEMP2A)	PWM output 2/MTU4.TGRA temporary register A	Not readable or writable
Temporary register 2B (TEMP2B)	PWM output 2/MTU4.TGRA temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 3A (TEMP3A)	PWM output 3/MTU4.TGRB temporary register A	Not readable or writable
Temporary register 3B (TEMP3B)	PWM output 3/MTU4.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 4A (TEMP4A)	PWM output 4/MTU6.TGRB temporary register A	Not readable or writable
Temporary register 4B (TEMP4B)	PWM output 4/MTU6.TGRB temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 5A (TEMP5A)	PWM output 5/MTU7.TGRA temporary register A	Not readable or writable
Temporary register 5B (TEMP5B)	PWM output 5/MTU7.TGRA temporary register B (when double buffer function is used)	Not readable or writable
Temporary register 6A (TEMP6A)	PWM output 6/MTU7.TGRB temporary register A	Not readable or writable
Temporary register 6B (TEMP6B)	PWM output 6/MTU7.TGRB temporary register B (when double buffer function is used)	Not readable or writable

Note 1. Access can be enabled or disabled according to the setting in TRWERA (timer read/write enable register A).

Note 2. Access can be enabled or disabled according to the setting in TRWERB (timer read/write enable register B).

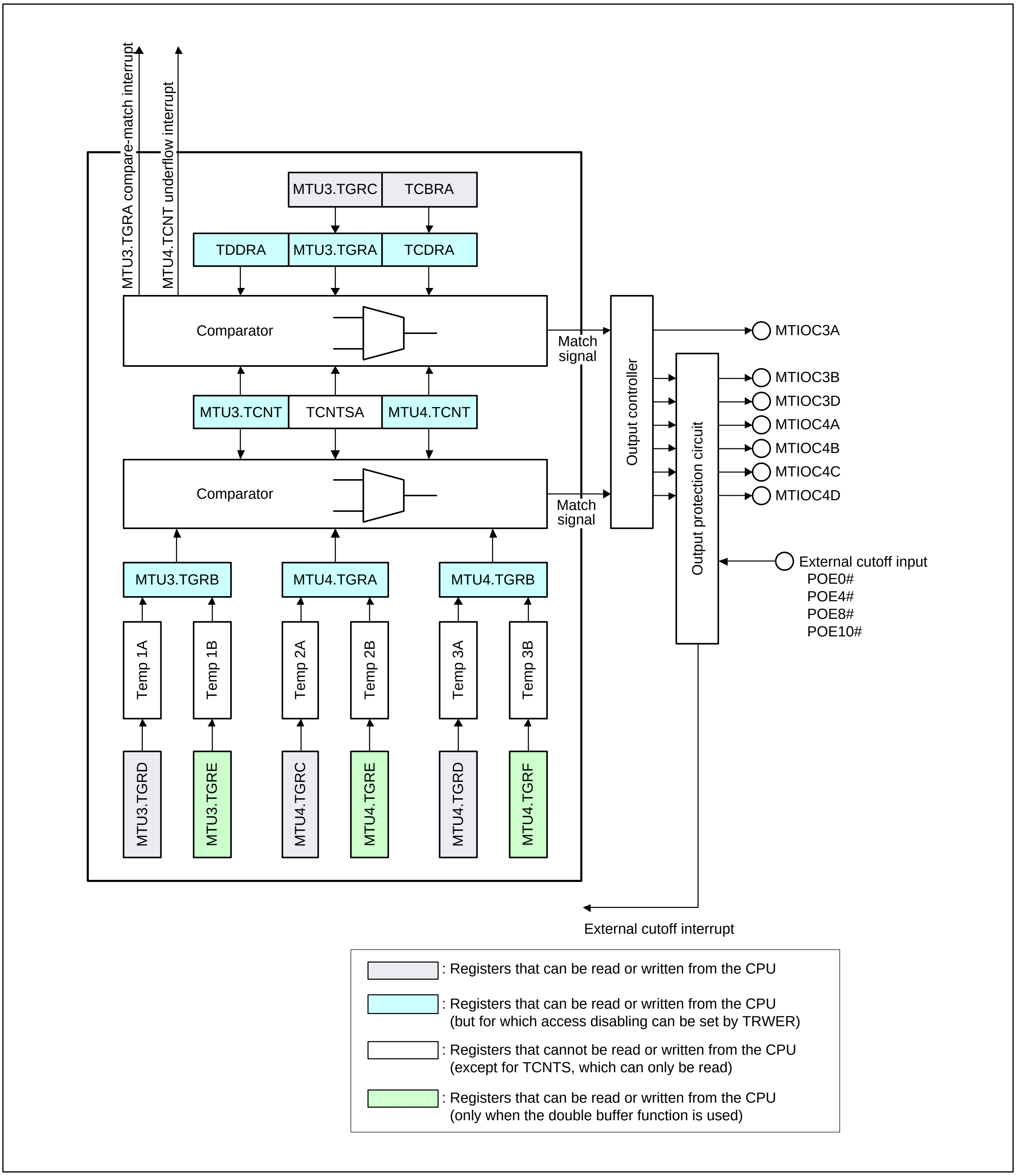


Figure 16.46 Block Diagram of MTU3 and MTU4 in Complementary PWM Mode

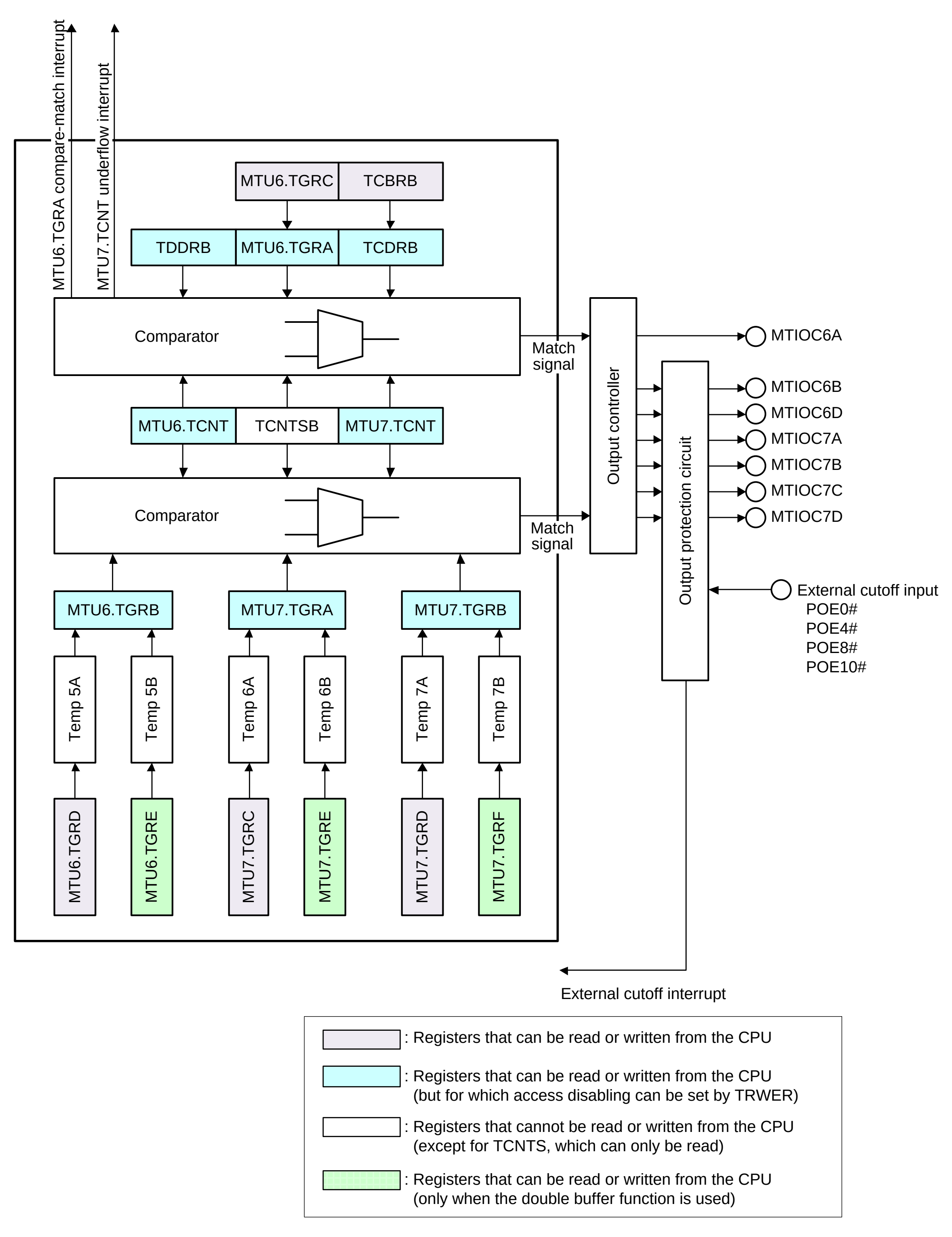
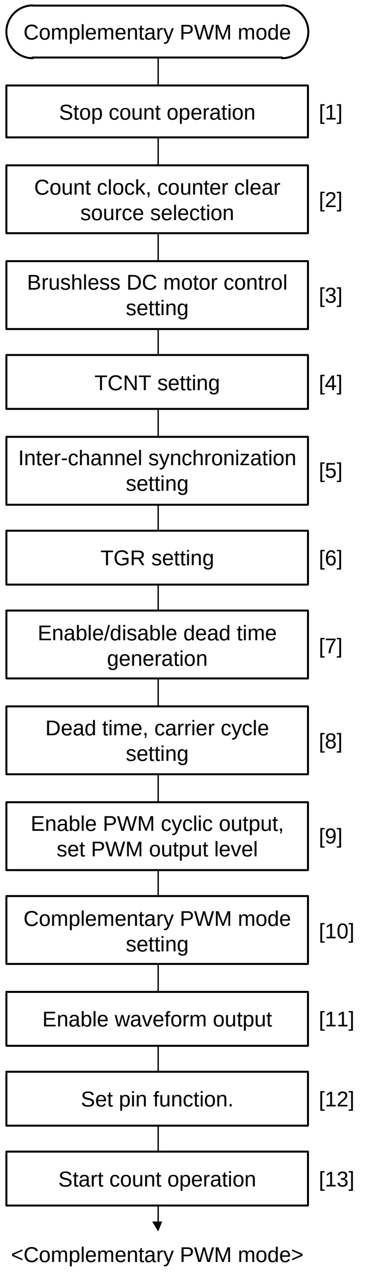


Figure 16.47 Block Diagram of MTU6 and MTU7 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

Figure 16.48 shows an example of the complementary PWM mode setting procedure.



[1] Clear the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 0 to stop the TCNT count operation.
Specify complementary PWM mode while MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT) are stopped.

[2] Set the TCR.TPSC[2:0] and TCR.CKEG[1:0] bits of MTU3 and MTU4 (MTU6 and MTU7) to select the same count clock source and clock edge for MTU3 and MTU4 (MTU6 and MTU7).
Set the TCR.CCLR[2:0] bits only when restarting by synchronous clearing with another channel during complementary PWM operation.

[3] To control a brushless DC motor, set the TGCRA.BDC bit and set the feedback signal input source and output chopping or gate signal direct output (only when MTU3 and MTU4 are used).

[4] Set the dead time in MTU3.TCNT (MTU6.TCNT). Set MTU4.TCNT (MTU7.TCNT) to H'0000.

[5] Make this setting only when restarting by synchronous clearing with another channel during complementary PWM mode operation. In this case, set the TSYRA register (TSYRB) to synchronize the channel generating the synchronous clear with MTU3 and MTU4 (MTU6 and MTU7).

[6] When MTU3 and MTU6 are used, set the PWM output duty ratio in the duty registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB) and buffer registers (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD).
Be sure to set the same value in each pair of the duty registers (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD) and buffer registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB).
Only when the double buffer function is used, set the PWM output duty value − 1 in the buffer registers (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF).
When MTU6 and MTU7 are used, set the value to the following registers:
Duty registers (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB)
Buffer registers (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD)
Buffer registers when the double buffer function is used (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF)

[7] This setting is necessary only when no dead time should be generated. Make appropriate settings in the TDERA (TDERB) register so that no dead time generated.

[8] Set the dead time in the TDDRA (TDDRB) register, 1/2 the PWM cycle in the TCDRA (TCDRB) register and TCBRA (TCBRB) register, and 1/2 the PWM cycle plus the dead time in MTU3.TGRA (MTU6.TGRA) and MTU3.TGRC (MTU6.TGRC).
When no dead time generation is selected, set 1 in the TDDRA (TDDRB) register and 1/2 the PWM cycle + 1 in MTU3.TGRA (MTU6.TGRA) and MTU3.TGRC (MTU6.TGRC).

[9] Set the PSYE bit in the TOCR1A register to enable or disable toggle output synchronized with the PWM cycle, and set the OLSP and OLSN bits to select the PWM output level. When specifying the PWM output level using the TOCR2A (TOCR2B) and TOLBRA (TOLBRA) in buffer operation, see Timer Output Level Buffer Registers (TOLBRA and TOLBRB).

[10] Select complementary PWM mode in MTU3.TMDR1 (MTU6.TMDR1). Specify the buffer operation (BFA = 1) of MTU3.TGRA (MTU6.TGRA) and the buffer operation (BFB = 1) of MTU3.TGRB (MTU6.TGRB).
Only when the double buffer function is used, set the TMDR2A (TMDR2B) register to enable the double buffer function.

[11] Set the TOERA (TOERB) register to enable or disable output from the PWM output pin. Then, set the MTU3.TIORH (MTU6.TIORH), MTU3.TIORL (MTU6.TIORL), MTU4.TIORH (MTU7.TIORH), and MTU4.TIORL (MTU7.TIORL) to H'00.

[12] Set the port mode registers (PMn) and Pmn port function control register (PFCm) of the GPIO ports.

[13] Set the TSTRA.CST3 (TSTRB.CST6) and TSTRA.CST4 (TSTRB.CST7) bits to 1 simultaneously to start count operation.

Figure 16.48 Example of Complementary PWM Mode Setting Procedure

(2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. **Figure 16.49** illustrates counter operation in complementary PWM mode (MTU3 and MTU4), and **Figure 16.50** shows an example of operation in complementary PWM mode.

(a) Counter Operation

In complementary PWM mode, three counters — MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) — in each unit perform up-/down-count operations.

MTU3.TCNT (MTU6.TCNT) is automatically initialized to the value set in TDDRA (TDDRB) when complementary PWM mode is selected and the CST3 bit in TSTRA (TSTRB) is 0. When the CST3 bit is set to 1, MTU3.TCNT (MTU6.TCNT) counts up to the value set in MTU3.TGRA (MTU6.TGRA), then switches to down-counting when it matches MTU3.TGRA (MTU6.TGRA). When the MTU4.TCNT (MTU7.TCNT) value matches H’0000, MTU3.TCNT (MTU6.TCNT) switches to up-counting, and the operation is repeated in this way.

TCNT in MTU4 (MTU7) should be initialized to H’0000 after a reset. When the CST4 bit is set to 1, MTU4.TCNT (MTU7.TCNT) counts up in synchronization with MTU3.TCNT (MTU6.TCNT), and switches to down-counting when MTU3.TCNT (MTU6.TCNT) matches MTU3.TGRA (MTU6.TGRA). On reaching H’0000, MTU4.TCNT (MTU7.TCNT) switches to up-counting, and the operation is repeated in this way. TCNTSA (TCNTSB) is a read-only counter. It does not need to be initialized after a reset.

In counting up by MTU3.TCNT and MTU4.TCNT (or MTU6.TCNT and MTU7.TCNT), MTU3.TCNT (or MTU6.TCNT) starts counting up when it matches TCDRA (or TCDRB) and switches to counting down when it matches MTU3.TGRA (or MTU6.TGRA). Furthermore, when MTU4.TCNT (or MTU7.TCNT) matches TDDRA (or TDDRB), TCNTSA (or TCNTSB) is set to the value in MTU3.TGRA (or MTU6.TGRA) and counting is stopped.

When MTU4.TCNT (MTU7.TCNT) matches TDDRA (TDDRB) during down-counting of MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT), TCNTSA (TCNTSB) starts up-counting, and when MTU4.TCNT (MTU7.TCNT) matches H’0000, the operation switches to down-counting. When MTU3.TCNT (MTU6.TCNT) matches TCDRA (TCDRB), TCNTSA (TCNTSB) is cleared to H’0000 and stops counting.

TCNTSA (TCNTSB) is compared with the compare register and temporary register, in which the PWM duty is specified, only during the count operation.

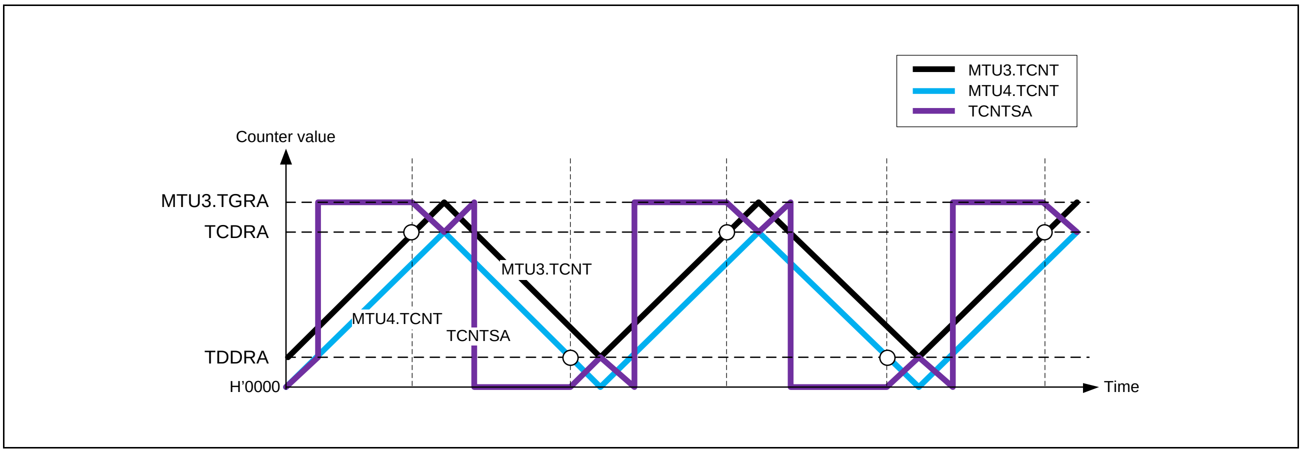


Figure 16.49 Count Operation in Complementary PWM Mode (MTU3 and MTU4)

(b) Register Operation

In complementary PWM mode, nine registers (compare registers, buffer registers, and temporary registers) are used to control the duty ratio for the PWM output. **Figure 16.50** shows an example of operation in complementary PWM mode (MTU3 and MTU4).

MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB) are constantly compared with the counters to generate PWM waveforms. When these registers match the counter, the value set in the OLSN and OLSP bits in the timer output control registers (TOCR1A and TOCR1B) is output from the PWM output pin. MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD) are buffer registers for these compare registers.

When the double buffer function is used, MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF) are also used as buffer registers B. For details of double buffer operation, refer to **Section 16.3.8(2)(s), Double Buffer Function in Complementary PWM Mode**.

Data in a compare register can be changed by writing new data to the corresponding buffer register. The buffer registers can be read or written at any time.

When modifying data in a buffer register, write to MTU4.TGRD (MTU7.TGRD) last and enable data transfer from the buffer register to a temporary register. At this time, transfer from the TCBRA (TCBRB) register and MTU3.TGRC (MTU6.TGRC) register, which operate as buffer registers for the timer cycle registers, to temporary registers is also enabled. Data is transferred to all five temporary registers at the same time.

When transfer is enabled in the Ta interval, data written to a buffer register is transferred to the temporary register. The data is not transferred to the temporary register in the Tb1 and Tb2 intervals. Data enabled for transfer in this interval is transferred to the temporary register at the end of this interval.

The value transferred to a temporary register is transferred to the compare register at the end of the Tb1 interval (when matches MTU3.TGRA (MTU6.TGRA) while TCNTSA (TCNTSB) is counting up), or at the end of the Tb2 interval (when matches H’0000 while TCNTSA (TCNTSB) is counting down). The timing for transfer from the temporary register to the compare register can be selected with bits MD[3:0] in the timer mode register 1 (TMDR1). **Figure 16.50** shows an example in which the trough is selected for the transfer timing.

In the Tb interval in which data is not transferred to the temporary register (Tb1 in **Figure 16.50**), the temporary register has the same function as the compare register and is compared with the counter. In this interval, therefore, there are two compare match registers for one output phase; the compare register contains the pre-change data and the temporary register contains new data. In this interval, three counters MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) and two registers (compare register and temporary register) are compared, and PWM output is controlled accordingly.

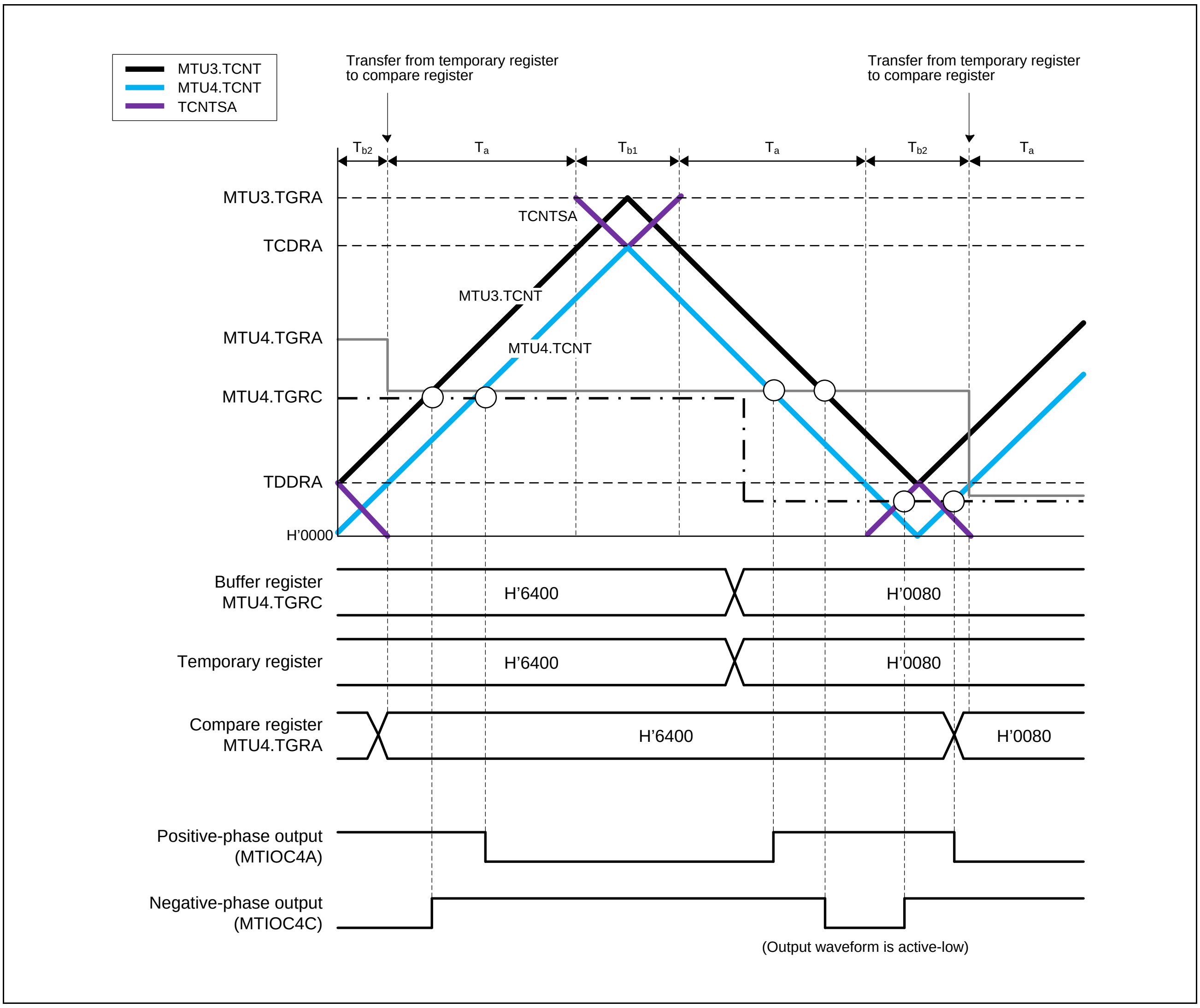


Figure 16.50 Example of Operation in Complementary PWM Mode (MTU3 and MTU4)

(c) Initial Setting

In complementary PWM mode, there are nine registers that require initial setting. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled).

Before setting complementary PWM mode with MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits, initial values should be set in the following registers.

The TOCR1A, TOCR2A, TOCR1B, and TOCR2B registers are used to set the PWM output level. MTU3.TGRC (MTU6.TGRC) operates as the buffer register for MTU3.TGRA (MTU6.TGRA), and should be set with 1/2 the PWM cycle + dead time Td. The timer cycle buffer register (TCBRA or TCBRB) operates as the buffer register for the timer cycle data register (TCDRA or TCDRB), and should be set with 1/2 the PWM cycle. Set dead time Td in the timer dead time data register (TDDRA or TDDRB).

When dead time is not needed, the TDER bit in the timer dead time enable register (TDERA or TDERB) should be cleared to 0, MTU3.TGRC and MTU3.TGRA (MTU6.TGRC and MTU6.TGRA) should be set to 1/2 the PWM carrier cycle + 1, and TDDRA (TDDRB) should be set to 1.

Set the respective initial PWM duty values in three buffer registers A (MTU3.TGRD, MTU4.TGRC, and MTU4.TGRD (MTU6.TGRD, MTU7.TGRC, and MTU7.TGRD)).

Set the respective (initial PWM duty – 1) values in three buffer registers B (MTU3.TGRE, MTU4.TGRE, and MTU4.TGRF (MTU6.TGRE, MTU7.TGRE, and MTU7.TGRF)) only when the double buffer function is used.

The values set in the five buffer registers excluding TDDRA (TDDRB) are transferred to the corresponding compare registers as soon as complementary PWM mode is set.

Set MTU4.TCNT (MTU7.TCNT) to H’0000 before setting complementary PWM mode.

Table 16.77 Registers and Counters Requiring Initial Setting

Register and Counter	Setting
TOCR1A, TOCR2A, TOCR1B, TOCR2B	PWM output level
MTU3.TGRC MTU6.TGRC	1/2 PWM cycle + dead time Td (1/2 PWM cycle + 1 when dead time generation is disabled by the TDERA or TDERB setting)
TDDRA, TDDRB	Dead time Td (1 when dead time generation is disabled by the TDERA or TDERB setting)
TCBRA, TCBRB	1/2 PWM cycle
MTU3.TGRD, MTU4.TGRC, MTU4.TGRD MTU6.TGRD, MTU7.TGRC, MTU7.TGRD	Initial PWM duty ratio value for each phase
MTU3.TGRE, MTU4.TGRE, MTU4.TGRF MTU6.TGRE, MTU7.TGRE, MTU7.TGRF	Initial PWM duty ratio – 1 value for each phase (only when double buffer function is used)
MTU4.TCNT MTU7.TCNT	H’0000

Note: The value set in MTU3.TGRC (MTU6.TGRC) should be the sum of 1/2 the PWM cycle set in TCBRA (TCBRB) and dead time Td set in TDDRA (TDDRB). When dead time generation is disabled by TDERA (TDERB), TGRC should be set to 1/2 the PWM cycle + 1.

(d) PWM Output Level Setting

In complementary PWM mode, the PWM output level is set with bits OLSN and OLSP in timer output control register 1 (TOCR1A or TOCR1B) or bits OLS1P to OLS3P and OLS1N to OLS3N in timer output control register 2 (TOCR2A or TOCR2B).

The output level can be set for each of the three positive phases and three negative phases of 6-phase output. Complementary PWM mode should be cleared before setting or changing output levels.

(e) Dead Time Setting

In complementary PWM mode, dead time can be set for PWM output.

The dead time is set in the timer dead time data register (TDDRA or TDDRB). The value set in TDDRA (TDDRB) is used as the MTU3.TCNT (MTU6.TCNT) counter start value and creates a non-overlapping interval between MTU3.TCNT (MTU6.TCNT) and MTU4.TCNT (MTU7.TCNT). Complementary PWM mode should be cleared before changing the contents of TDDRA (TDDRB).

(f) Dead Time Suppressing

Dead time generation is suppressed by clearing the TDER bit in the timer dead time enable register (TDERA or TDERB) to 0. TDERA (TDERB) can be cleared to 0 only when 0 is written to it after reading TDER = 1.

MTU3.TGRA and MTU4.TGRC (MTU6.TGRA and MTU7.TGRC) should be set to 1/2 PWM cycle + 1 and the timer dead time data register (TDDRA or TDDRB) should be set to 1.

By the above settings, PWM waveforms without dead time can be obtained. **Figure 16.51** shows an example of operation without dead time (MTU3 and MTU4).

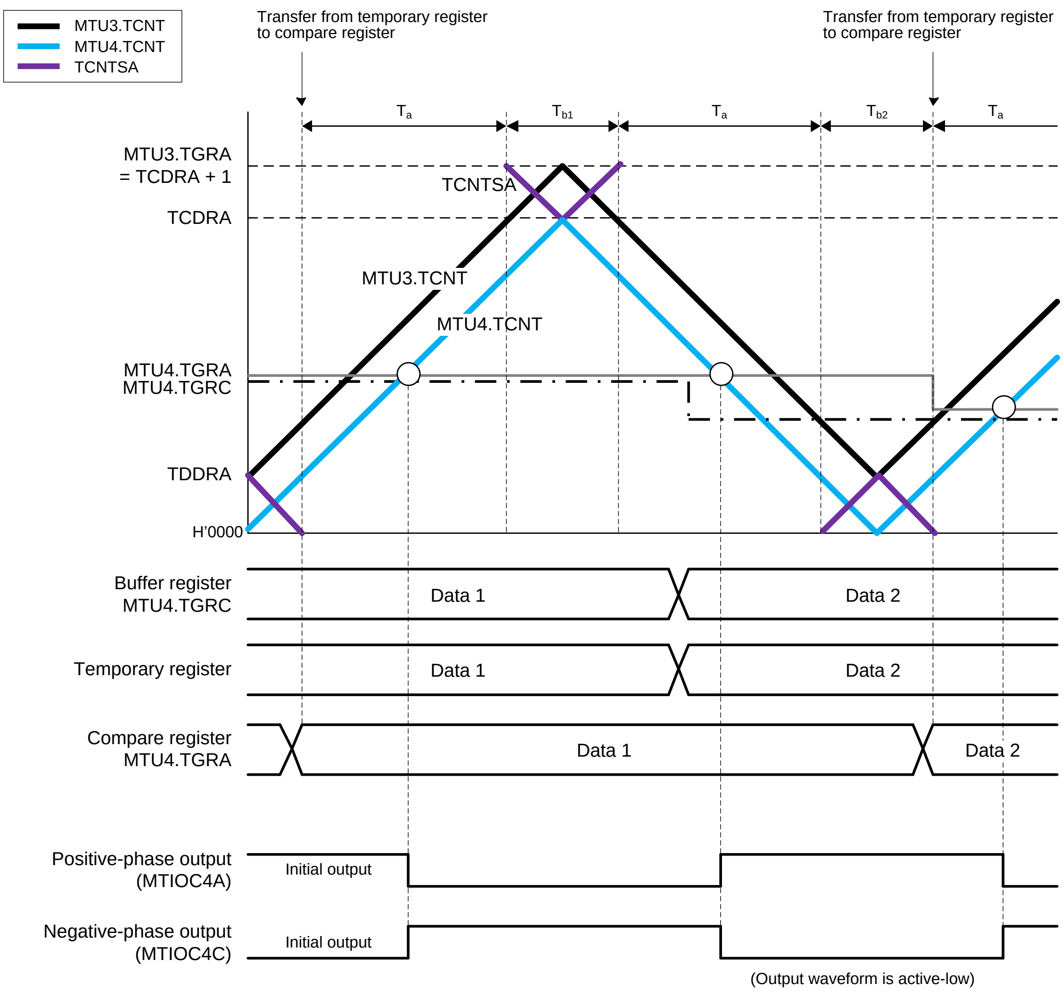


Figure 16.51 Example of Operation without Dead Time (MTU3 and MTU4)

(g) PWM Cycle Setting

In complementary PWM mode, the PWM cycle is set in two registers—MTU3.TGRA (MTU6.TGRA), in which the MTU3.TCNT (MTU6.TCNT) upper limit value is set, and TCDRA (TCDRB), in which the MTU4.TCNT (MTU7.TCNT) upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

With dead time: MTU3.TGRA (MTU6.TGRA) setting = TCDRA (TCDRB) setting + TDDRA (TDDRB) setting
Without dead time: MTU3.TGRA (MTU6.TGRA) setting = TCDRA (TCDRB) setting + 1

In addition, the settings should be made so as to achieve the following relationship between the TCDRA (TCDRB) register and the TDDRA (TDDRB) register:

TCDRA (TCDRB) setting > TDDRA (TDDRB) setting × 2 + 2

The MTU3.TGRA and TCDRA (MTU6.TGRA and TCDRB) settings are made by setting values in buffer registers MTU3.TGRC and TCBRA (MTU6.TGRC and TCBRB). When data is written to MTU4.TGRD (MTU7.TGRD) to enable transfers, the values set in MTU3.TGRC and TCBRA (MTU6.TGRC and TCBRB) are transferred simultaneously to the MTU3.TGRA and TCDRA (MTU6.TGRA and TCDRB) with the transfer timing selected with the MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits.

The new PWM cycle is reflected from the next cycle when data is updated at the crest, or from the current cycle when updated in the trough. **Figure 16.52** illustrates the operation when the PWM cycle is updated at the crest.

See the following **Section 16.3.8(2)(h), Register Data Updating**, for the method of updating the data in each buffer register.

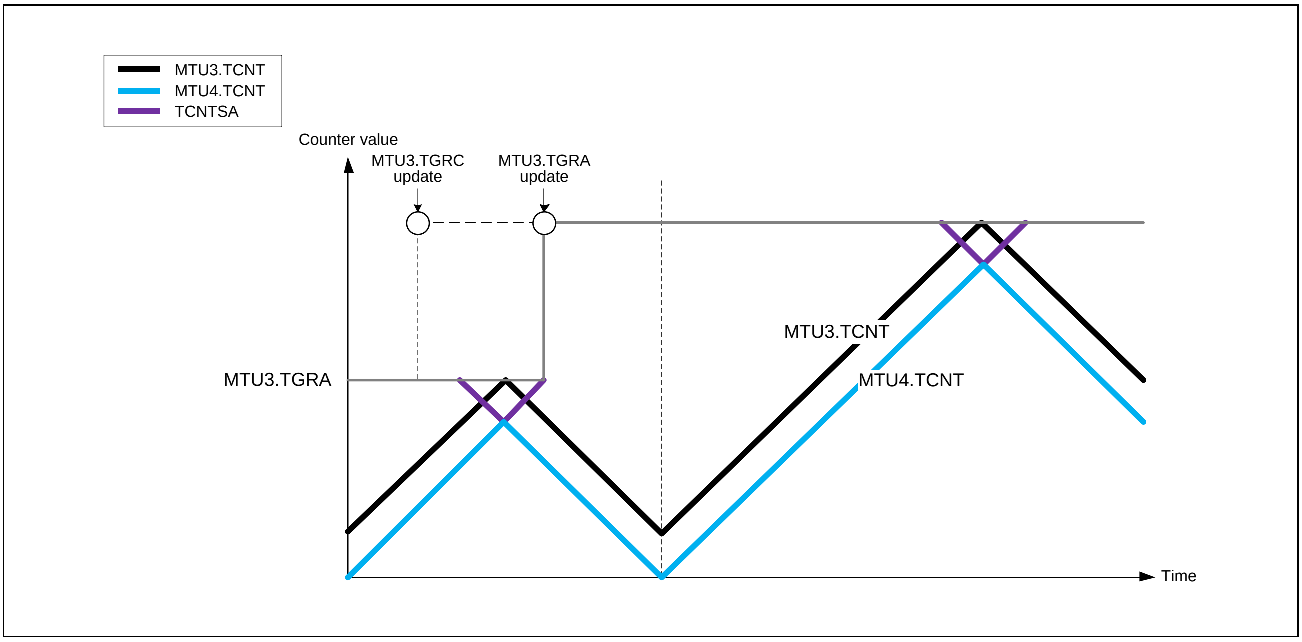


Figure 16.52 Example of PWM Cycle Updating (MTU3 and MTU4)

(h) Register Data Updating

In complementary PWM mode, buffer registers are used to update the data in five compare registers for PWM duty and PWM cycle. The update data can be written to the buffer registers at any time.

There is a temporary register between each of these registers and its buffer register. While subcounter TCNTSA (TCNTSB) is not counting, if buffer register data is updated, the temporary register value also changes. Data is not transferred from buffer registers to temporary registers while TCNTSA (TCNTSB) is counting; in this case, the value written to a buffer register is transferred after TCNTSA (TCNTSB) halts.

The temporary register value is transferred to the compare register at the data update timing set with MTU3.TMDR1.MD[3:0] (MTU6.TMDR1.MD[3:0]) bits. **Figure 16.53** shows an example of data updating in complementary PWM mode (MTU3 and MTU4). This example shows the mode in which data is updated at both the counter crest and trough.

When updating buffer register data, be sure to write to MTU4.TGRD (MTU7.TGRD) at the end of the update. Data is transferred from buffer registers to the temporary registers simultaneously for all five registers after the write to MTU4.TGRD (MTU7.TGRD).

Even when not updating all five registers or when not updating the MTU4.TGRD (MTU7.TGRD) data, be sure to write to MTU4.TGRD (MTU7.TGRD) after writing data to the registers to be updated. In this case, the data written to MTU4.TGRD (MTU7.TGRD) should be the same as the data prior to the write operation.

See **Section 16.3.8(2)(s), Double Buffer Function in Complementary PWM Mode**, for data updating when the double buffer function is used.

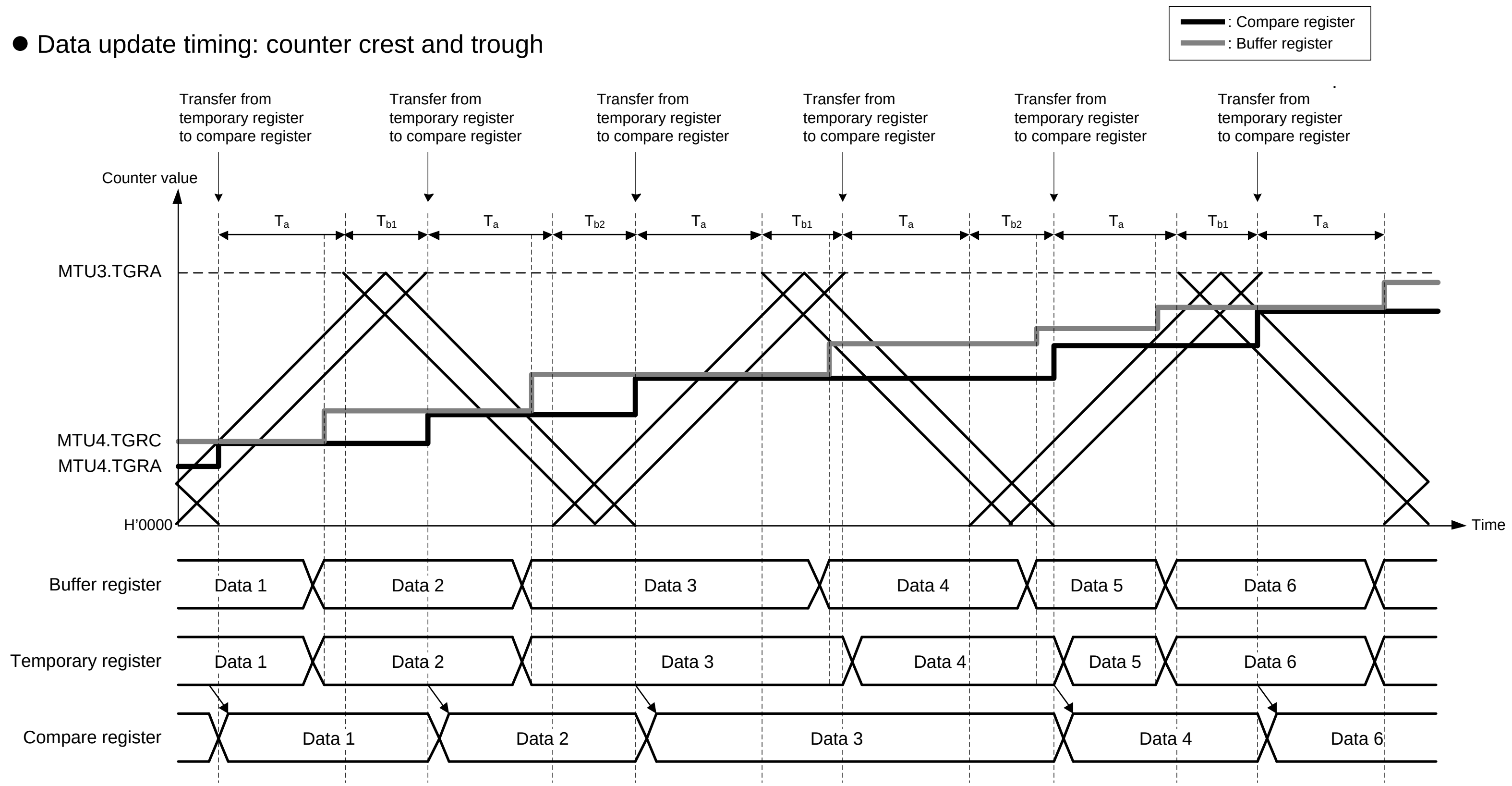


Figure 16.53 Example of Data Updating in Complementary PWM Mode (MTU3 and MTU4)

(i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of the OLSN and OLSP bits in the TOCR1A (TOCR1B) register or the OLS1N to OLS3N and OLS1P to OLS3P bits in the TOCR2A register (TOCR2B). This initial output is the non-active level of the PWM output and continues from when complementary PWM mode is set with the MTU3.TMDR1 (MTU6.TMDR1) until MTU4.TCNT (MTU7.TCNT) exceeds the value set in the TDDRA (TDDRB) register. **Figure 16.54** shows an example of the initial output in complementary PWM mode.

An example of the waveform when the initial PWM duty ratio value is smaller than the TDDRA (TDDRB) value is shown in **Figure 16.55**.

● Timer output control register settings

TOCR1A.OLSN bit = 0 (initial output: high; active level: low)
TOCR1A.OLSP bit = 0 (initial output: high; active level: low)

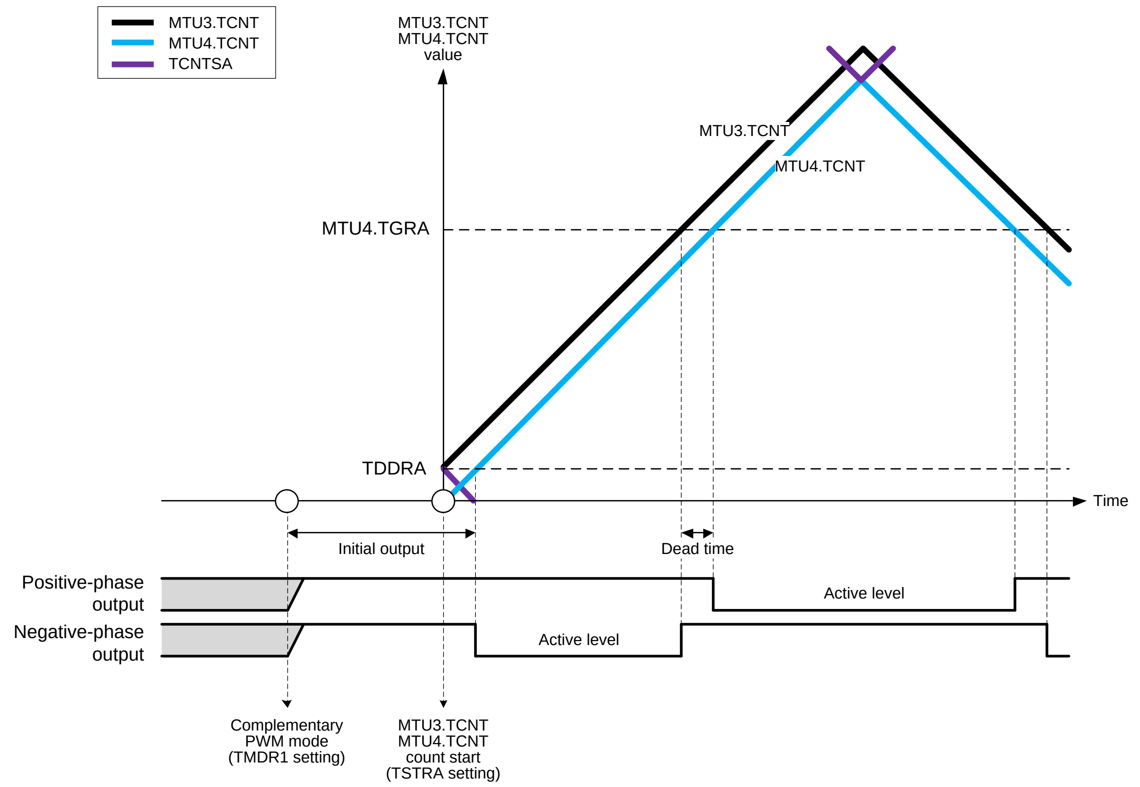


Figure 16.54 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (1)

- Timer output control register settings

TOCR1A.OLSN bit = 0 (initial output: high; active level: low)
TOCR1A.OLSP bit = 0 (initial output: high; active level: low)

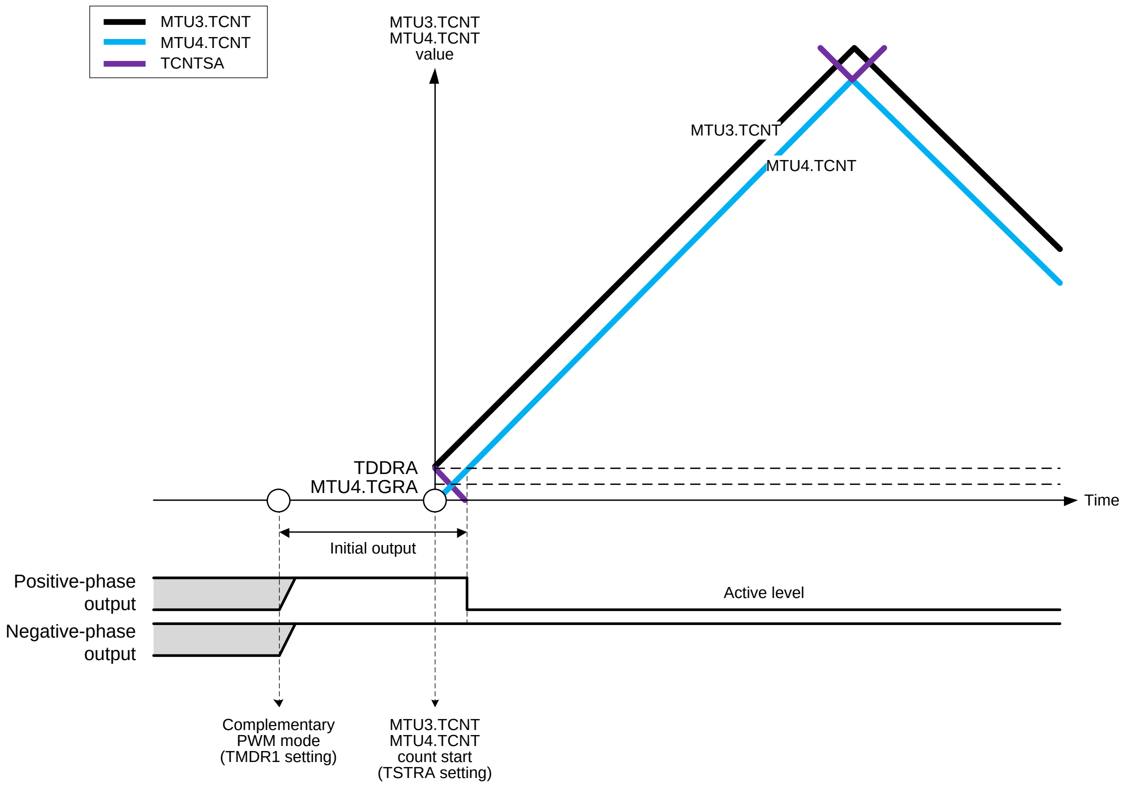


Figure 16.55 Example of Initial Output in Complementary PWM Mode (MTU3 and MTU4) (2)

(j) Method for Generating PWM Output in Complementary PWM Mode

In complementary PWM mode, six phases (three positive and three negative) PWM waveforms can be output. Dead time can be set for PWM waveforms to be output.

A PWM waveform is generated by output of the level selected in the timer output control register in the event of a compare match between a counter and a compare register. While TCNTSA (TCNTSB) is counting, the compare register and temporary register values are simultaneously compared to create consecutive PWM output from 0 to 100% duty ratio. The relative timing of turn-on and turn-off compare match occurrence may vary, but the compare match that turns off each phase takes precedence to secure the dead time and ensure that the positive-phase and negative-phase turn-on times do not overlap. **Figure 16.56** to **Figure 16.58** show examples of waveform generation in complementary PWM mode.

The positive-phase and negative-phase turn-off timing is generated by a compare match with the counter indicated by a solid line, and the turn-on timing is generated by a compare match with the counter indicated by a dotted line, which operates with a delay equal to the dead time behind the counter indicated by a solid line. In the T1 period, compare match a that turns off the negative phase has the highest priority, and compare matches before a are ignored. In the T2 period, compare match c that turns off the positive phase has the highest priority, and compare matches before c are ignored.

In most cases, compare matches occur in the order a → b → c → d (or c → d → a' → b') as shown in **Figure 16.56**.

If compare matches deviate from the a → b → c → d order, since the time for which the negative phase is off is shorter than twice the dead time, the positive phase is not turned on. If compare matches deviate from the c → d → a' → b' order, since the time for which the positive phase is off is shorter than twice the dead time, the negative phase is not turned on. As shown in **Figure 16.57**, if compare match c follows compare match a before compare match b, compare match b is ignored and the negative phase is turned on by compare match d. This is because turning off the positive phase has priority due to the occurrence of compare match c (positive-phase off timing) before compare match b (positive-phase on timing) (consequently, the waveform does not change because the positive phase goes from off to off).

Similarly, in the example in **Figure 16.58**, turning off the negative phase has priority due to the occurrence of compare match a' (negative-phase off timing) before compare match d (negative-phase on timing). As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare matches at turn-off timings take precedence, and turn-on timing compare matches that occur before a turn-off timing compare match are ignored.

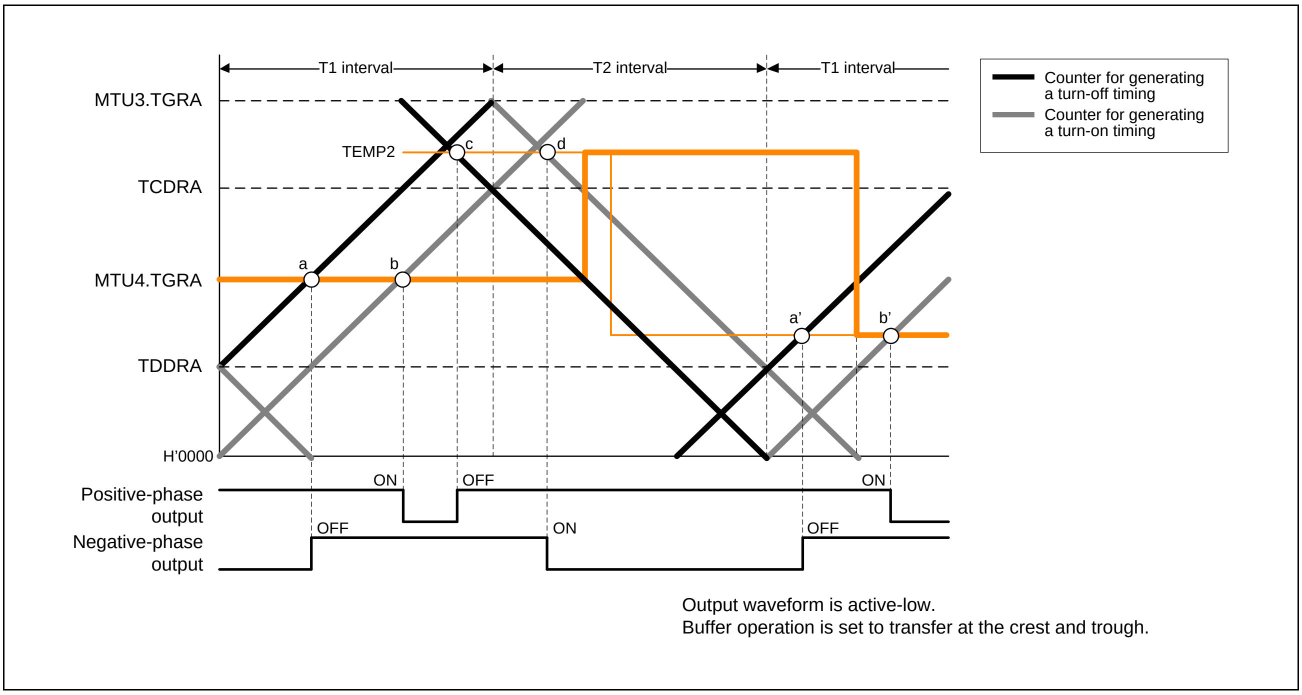


Figure 16.56 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

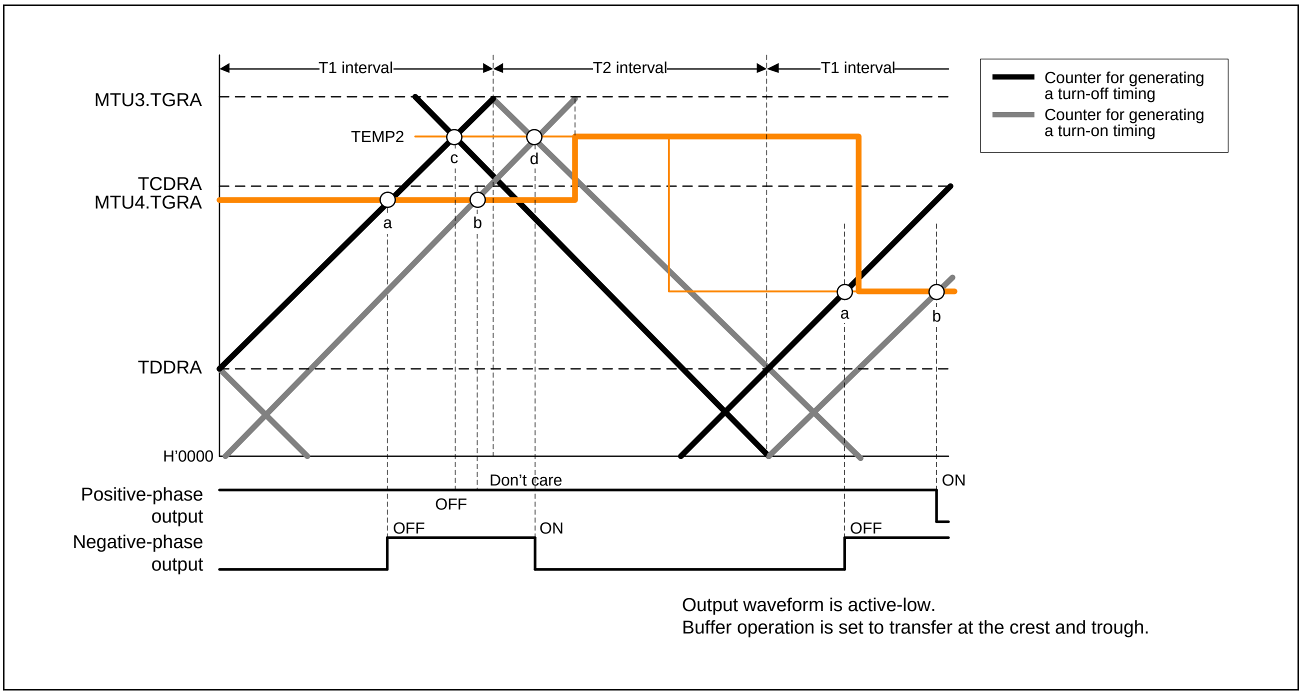


Figure 16.57 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

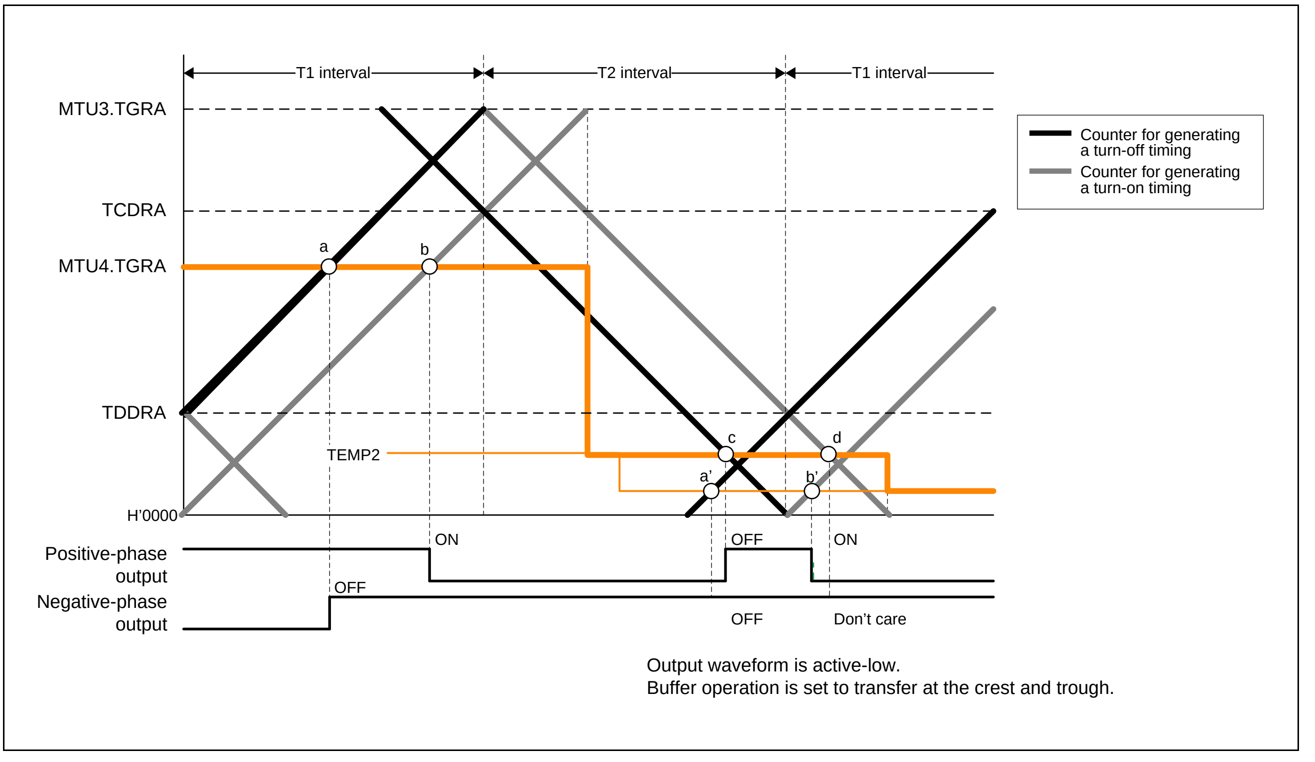


Figure 16.58 Example of Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

(k) 0% and 100% Duty Ratio Output in Complementary PWM Mode

In complementary PWM mode, 0% and 100% duty PWM output can be output as required. **Figure 16.59** to **Figure 16.63** show output examples.

A 100% duty waveform is output when the compare register value is set to H'0000. The waveform in this case has a positive phase with a 100% on-state. A 0% duty waveform is output when the compare register value is set to the same value as MTU3.TGRA (MTU6.TGRA). The waveform in this case has a positive phase with a 100% off-state.

Turn-on and turn-off compare matches occur simultaneously, but if a turn-on compare match and turn-off compare match for the same phase occur simultaneously, both compare matches are ignored and the waveform does not change.

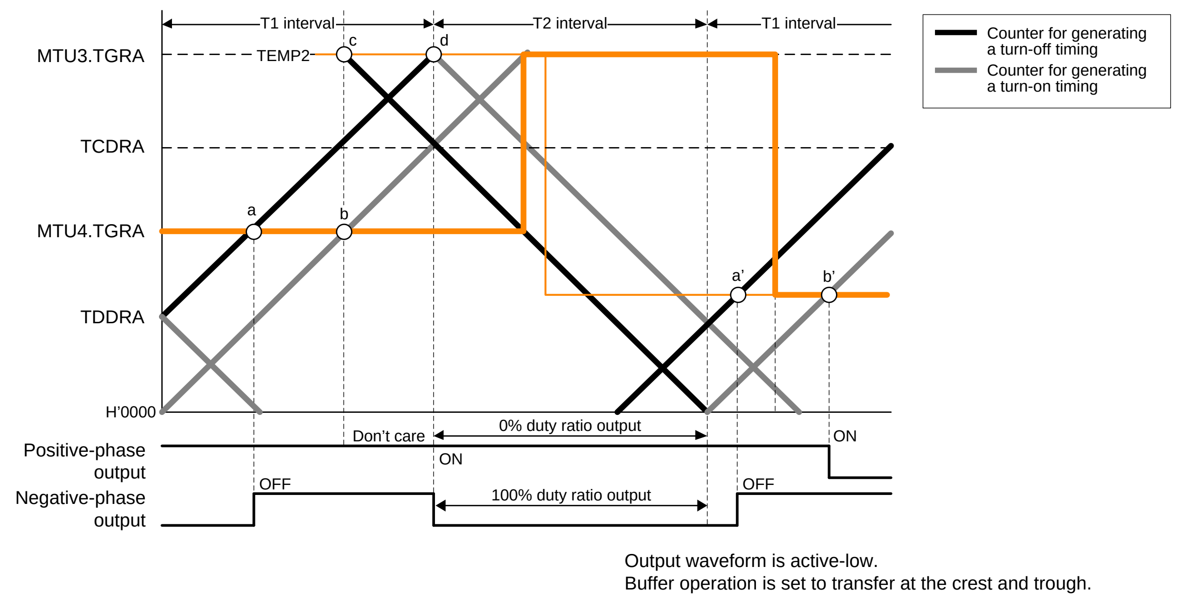


Figure 16.59 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (1)

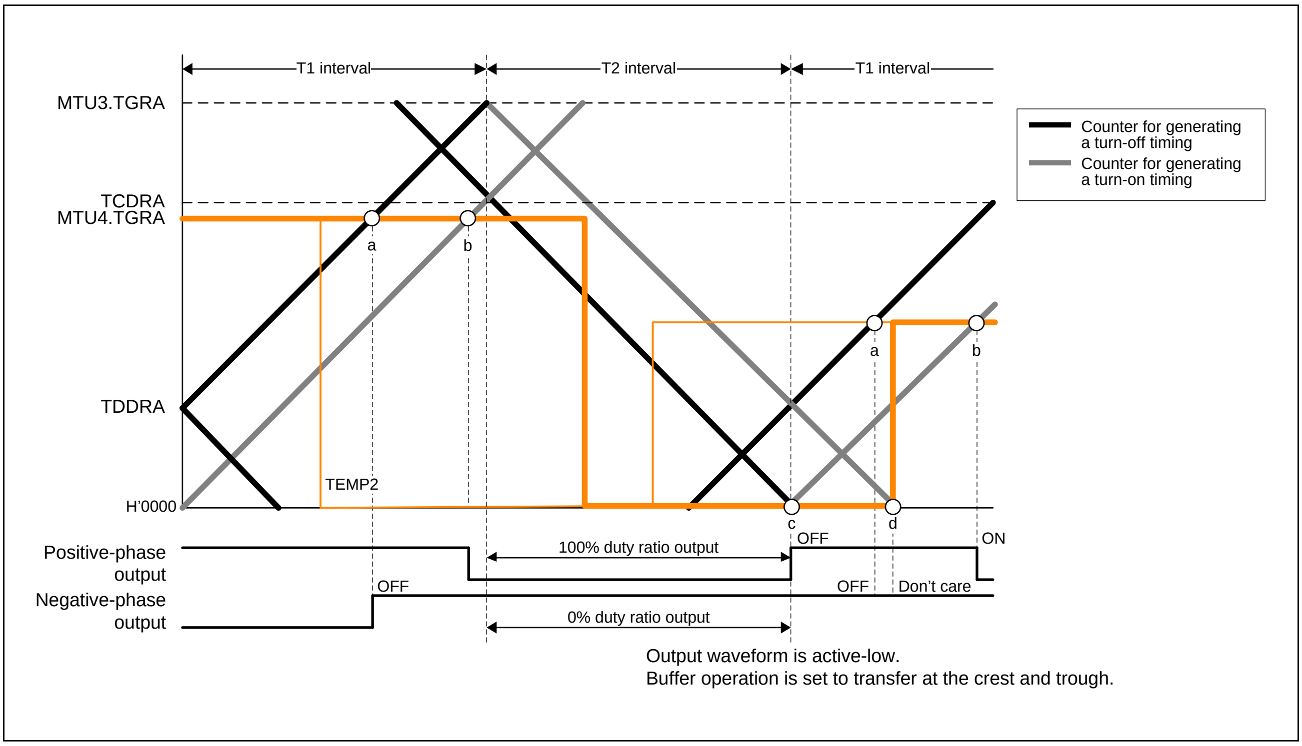


Figure 16.60 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (2)

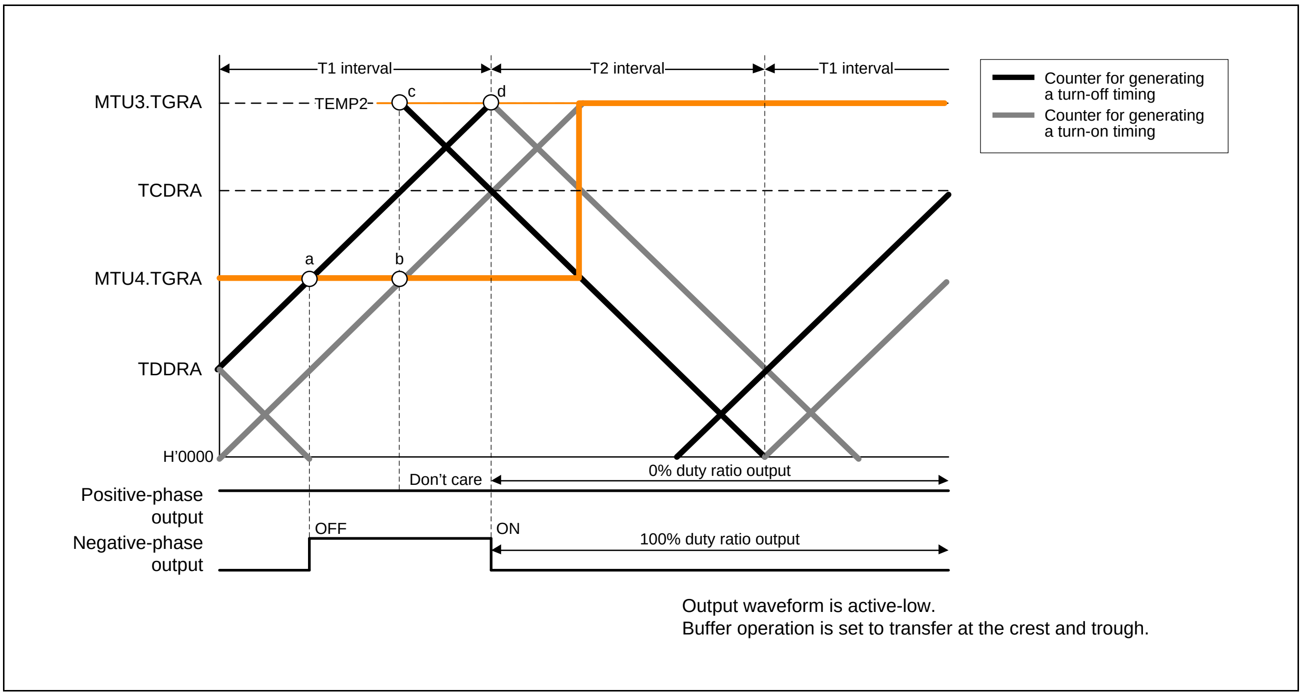


Figure 16.61 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (3)

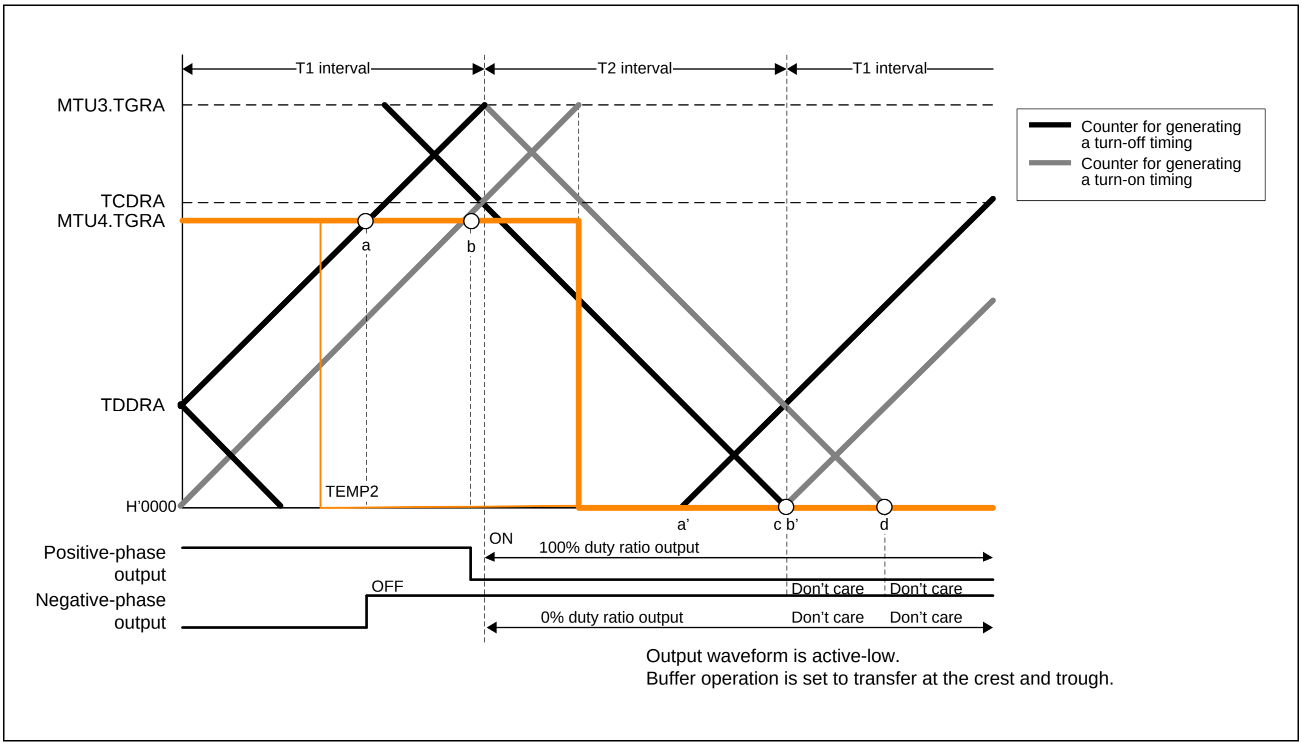


Figure 16.62 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (4)

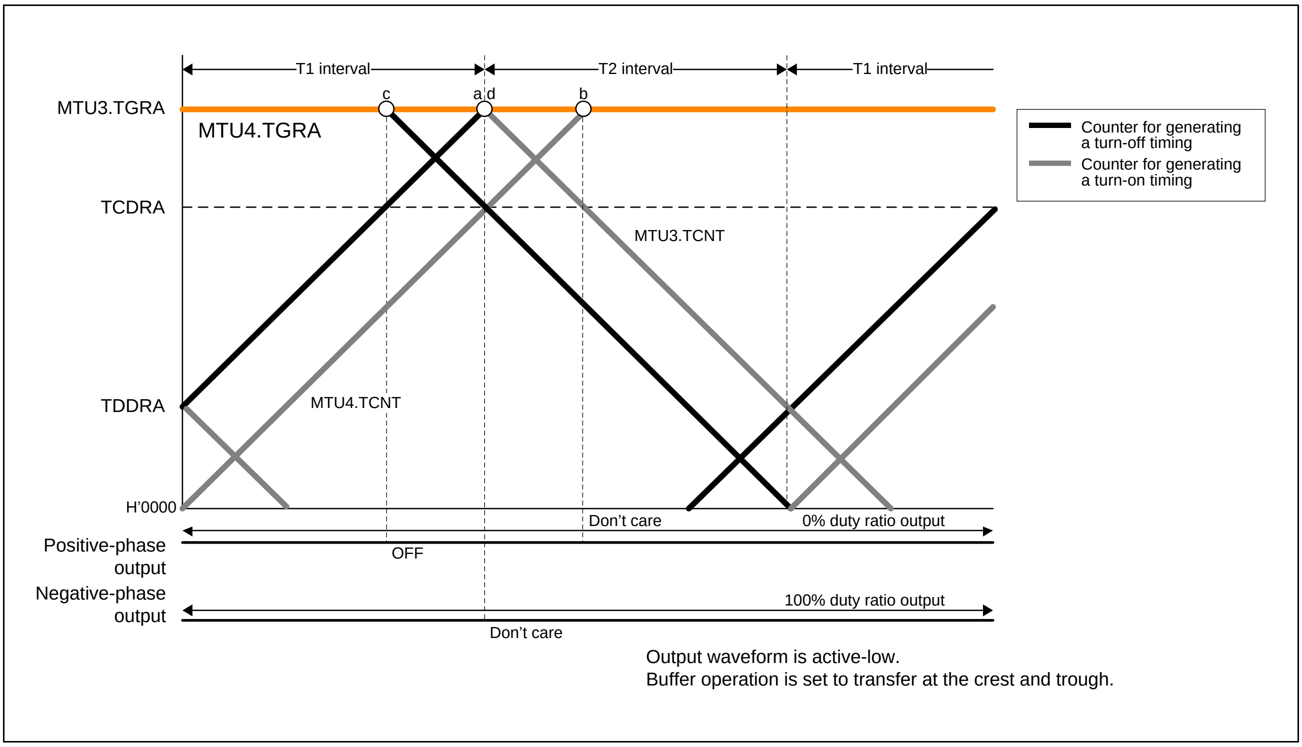


Figure 16.63 Example of 0% and 100% Waveform Output in Complementary PWM Mode (MTU3 and MTU4) (5)

(l) Toggle Output Synchronized with PWM Cycle

In complementary PWM mode, toggle output from the PWM output pin in synchronization with the PWM cycle can be enabled by setting the PSYE bit in the TOCR1A (TOCR1B) register to 1. An example of a toggle output waveform is shown in **Figure 16.64**.

This output is toggled by a compare match between MTU3.TCNT and MTU3.TGRA (MTU6.TCNT and MTU6.TGRA) and a compare match between MTU4.TCNT (MTU7.TCNT) and H’0000.

The MTIOC3A (MTIOC6A) pin is assigned for this toggle output. The initial output is high-level output.

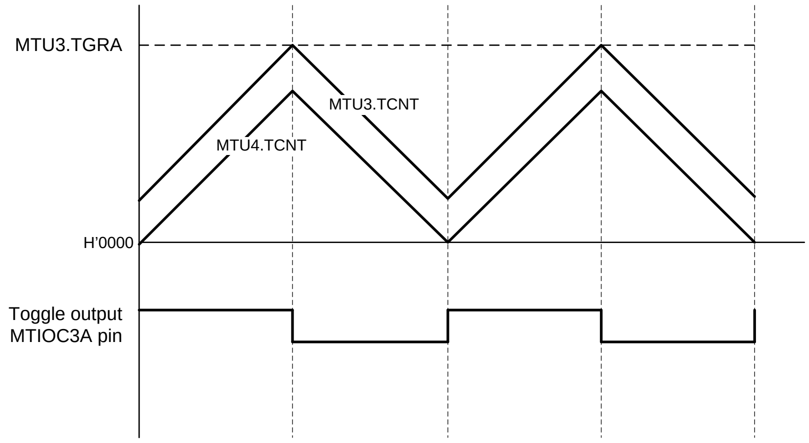


Figure 16.64 Example of Toggle Output Waveform Synchronized with PWM Output (MTU3 and MTU4)

(m) Counter Clearing by Another Channel

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) can be cleared by another channel source when a mode for synchronization with another channel is specified through the TSYRA (TSYRB) register and synchronous clearing is selected with MTU3.TCR.CCLR[2:0] (MTU6.TCR.CCLR[2:0]) bits.

Figure 16.65 illustrates an example of this operation.

Use of this function enables a counter to be cleared and restarted through an external signal.

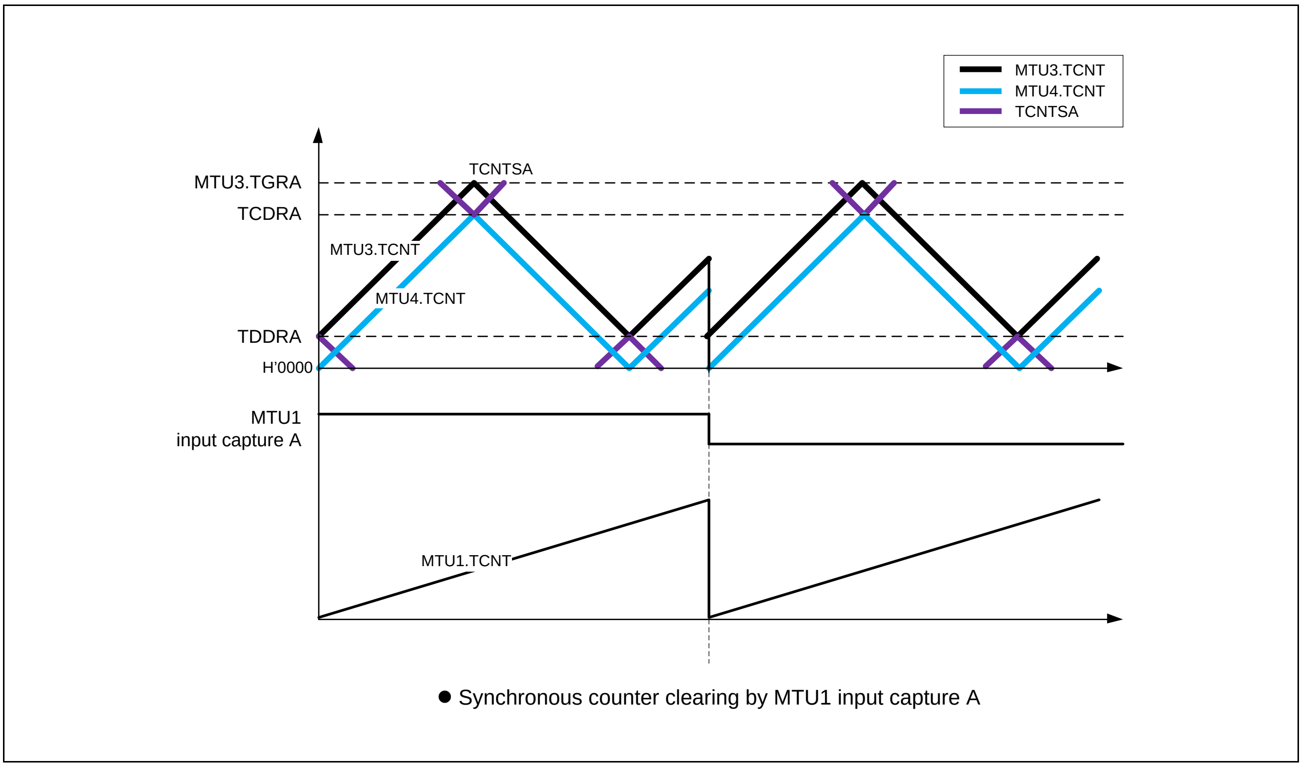


Figure 16.65 Counter Clearing Synchronized with Another Channel (MTU3 and MTU4)

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in TWCRA (TWCRB) to 1 suppresses initial output when synchronous counter clearing occurs in the Tb interval (Tb2 interval) at the trough in complementary PWM mode and controls abrupt change in duty cycle at synchronous counter clearing.

Initial output suppression through the WRE bit = 1 is applicable only when synchronous clearing occurs in the Tb2 interval as indicated by (10) or (11) in **Figure 16.66**. When synchronous clearing occurs outside that interval, the initial value specified by the OLSN and OLSP bits in TOCR1A (TOCR1B) is output. Even in the Tb2 interval, if synchronous clearing occurs in the initial value output period (indicated by (1) in **Figure 16.66**) immediately after the counters start operation, initial value output is not suppressed.

This function can be used in both channel combinations of MTU 3 and MTU4 and MTU6 and MTU7. In MTU3 and MTU4, synchronous clearing in MTU0, MTU1, and MTU2 can cause counter clearing; in MTU6 and MTU7, compare match or input capture generated in MTU0, MTU1, and MTU2 can cause counter clearing.

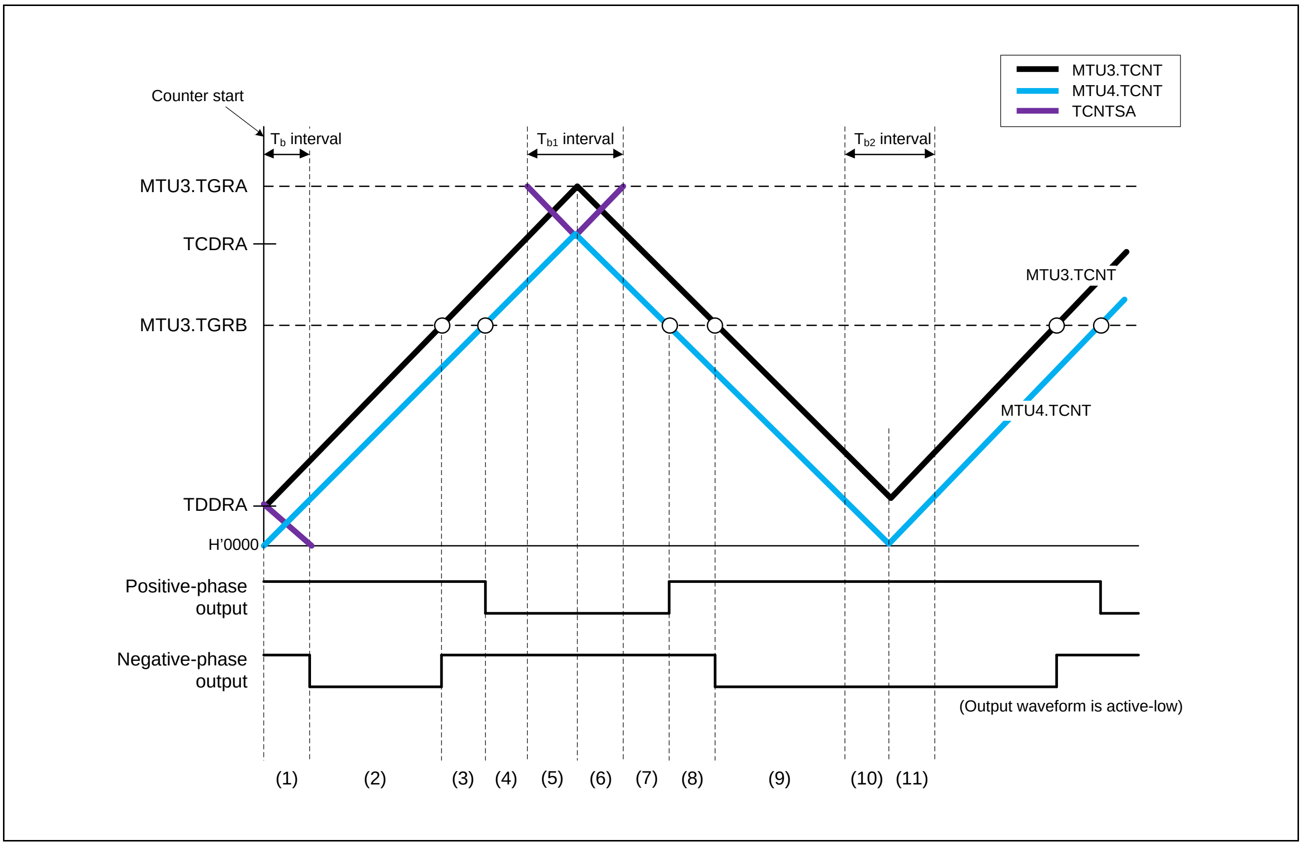
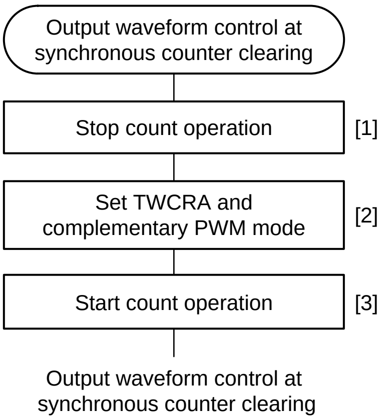


Figure 16.66 Timing for Synchronous Counter Clearing (MTU3 and MTU4)

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode.

  An example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode is shown in **Figure 16.67**.



[1] Clear bits CST3 and CST4 in TSTRA to 0 to stop count operation. Set up the TWCRA register while the TCNT counters in MTU3 and MTU4 are stopped.

[2] Read bit WRE in TWCRA and then write 1 to it to suppress initial value output at counter clearing.

[3] Set bits CST3 and CST4 in TSTRA to 1 at the same time to start count operation.

Figure 16.67 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode (MTU3 and MTU4)

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Figure 16.68 to **Figure 16.71** show examples of output waveform control in which MTU3 and MTU4 operate in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in TWCRA is set to 1. In the examples shown in **Figure 16.68** to **Figure 16.71**, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in **Figure 16.66**, respectively.
In MTU6 and MTU7, these examples are equivalent to the cases when MTU6 and MTU7 operate in complementary PWM mode and synchronous counter clearing is generated while the SCC bit is cleared to 0 and the WRE bit is set to 1 in TWCRB.

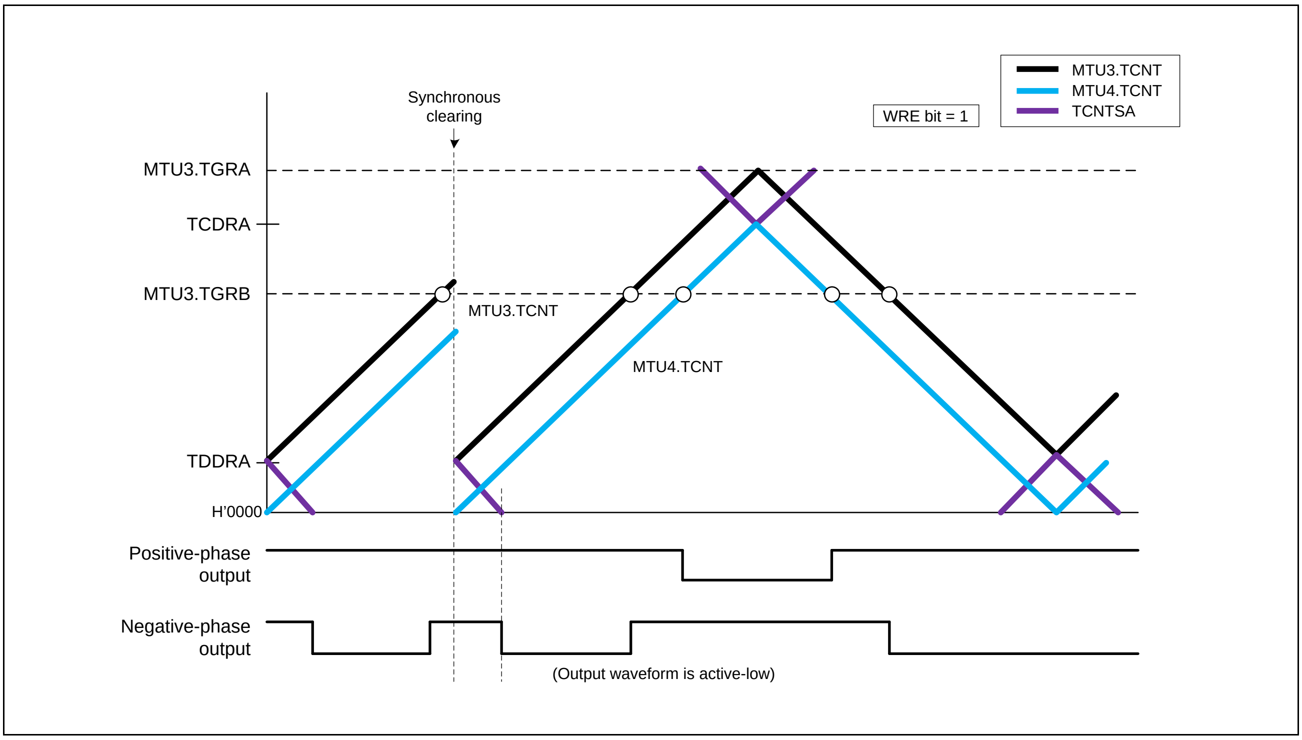


Figure 16.68 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in **Figure 16.66**; TWCRA.WRE Bit is 1)

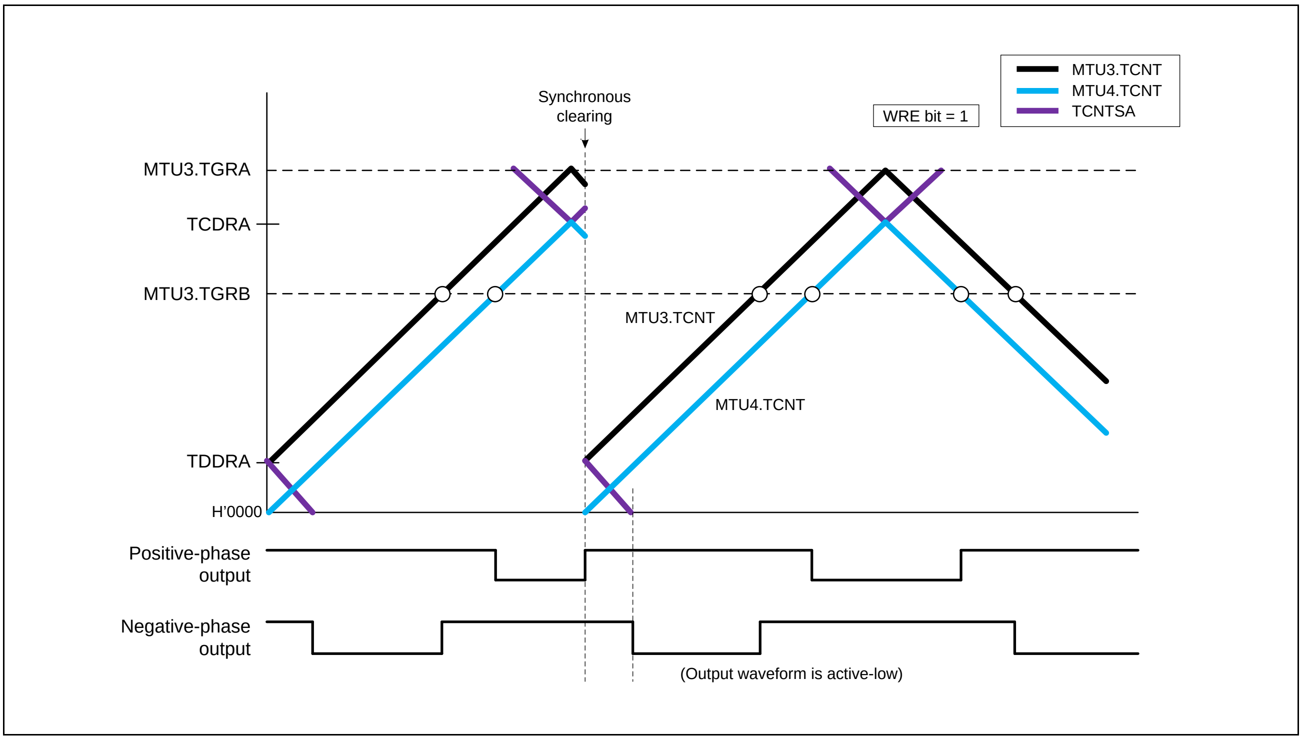


Figure 16.69 Example of Synchronous Clearing in Tb1 interval (Timing (6) in **Figure 16.66**; TWCRA.WRE Bit is 1)

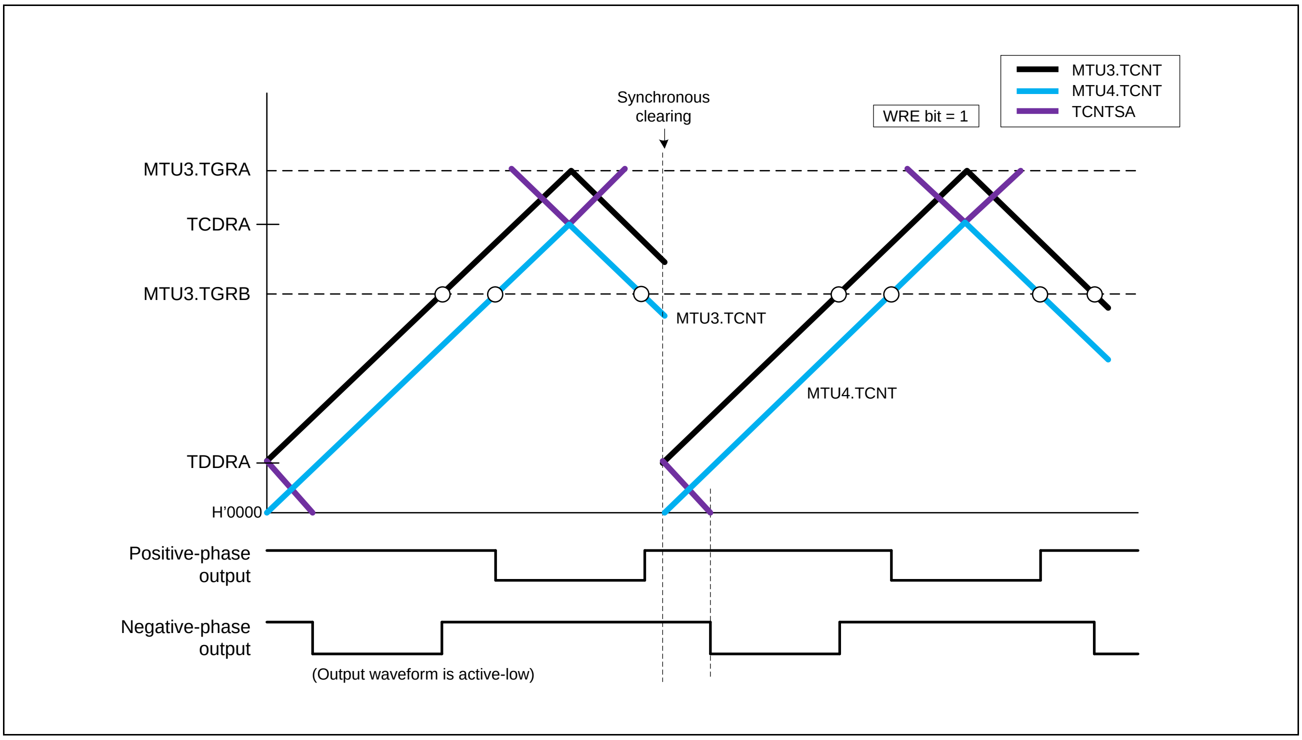


Figure 16.70 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in **Figure 16.66**; TWCRA.WRE Bit is 1)

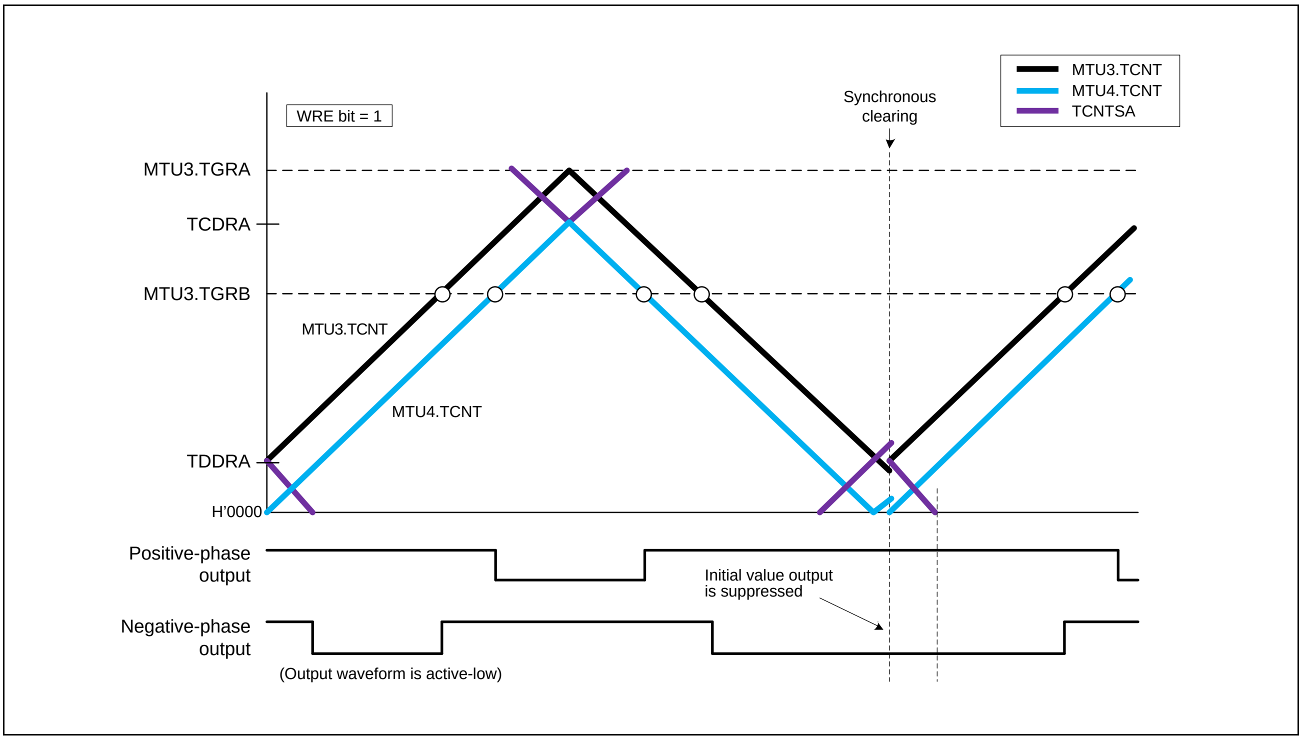


Figure 16.71 Example of Synchronous Clearing in Tb2 interval (Timing (11) in **Figure 16.66**; TWCRA.WRE Bit is 1)

(o) Suppressing Synchronous Counter Clearing for MTU0 to MTU2 and MTU6 and MTU7

In MTU6 and MTU7, setting the SCC bit in TWCRB to 1 suppresses synchronous counter clearing caused by MTU0 to MTU2.

Synchronous counter clearing is suppressed only within the interval shown in **Figure 16.72**. When using this function, MTU6 and MTU7 should be set to complementary PWM mode.

For details of synchronous clearing caused by MTU0 to MTU2, refer to **Section 16.3.10(2), Synchronous Counter Clearing for MTU6 and MTU7**.

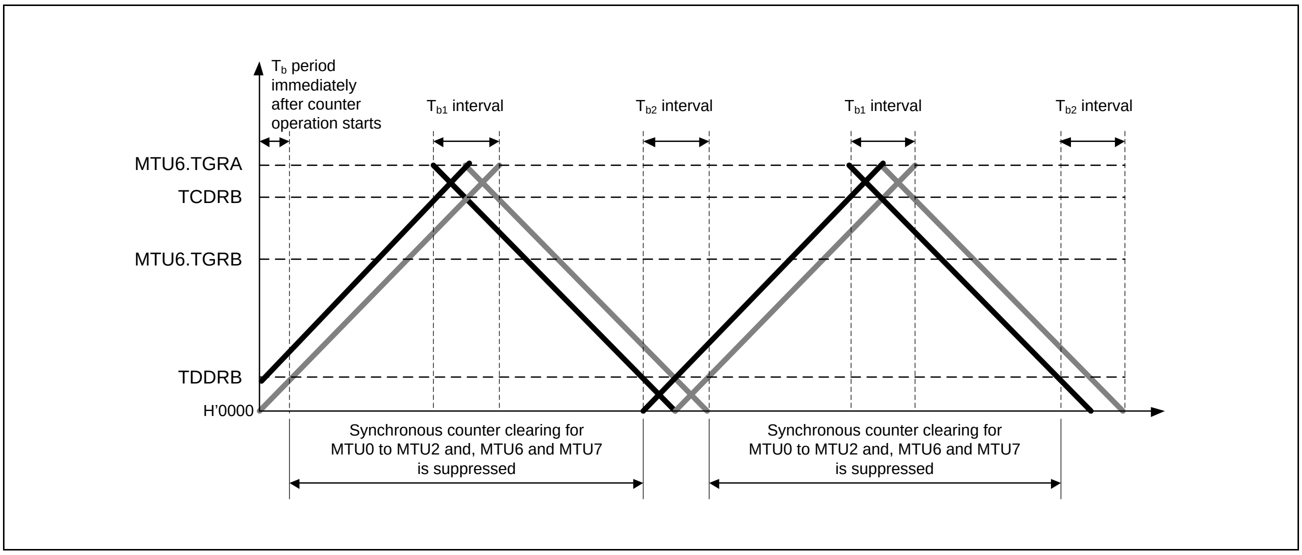
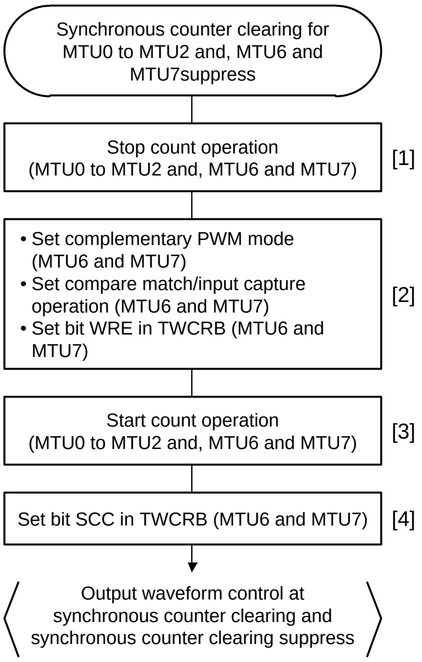


Figure 16.72 Synchronous Clearing-Suppressed Interval Specified by TWCRB.SCC Bit for MTU0, MTU1 and MTU2, and MTU6 and MTU7

- Example of Procedure for Suppressing Synchronous Counter Clearing for MTU0 to MTU2, and MTU6 and MTU7

An example of the procedure for suppressing synchronous counter clearing for MTU0 to MTU2, and MTU6 and MTU7 is shown in **Figure 16.73**.



[1] Clear bits CSTn of the timer start register (TSTRB) in MTU6 and MTU7 to 0 to stop count operation. Clear bits CSTn of TSTRA in MTU0 to MTU2 to 0 to stop count operation.

[2] Set the complementary PWM mode in MTU6 and MTU7 and compare match/input capture operation in MTU0 to MTU2. When bit WRE in TWCRA or TWCRB should be set, make appropriate setting here.

[3] Set bits CST6 and CST7 in MTU6 and MTU7 to 1 to start count operation. For synchronous counter clearing for MTU0 to MTU2 and MTU6 and MTU7, set bits CSTn of TSTRA in MTU0 to MTU2 to 1 to start count operation in any one of MTU0.TCNT to MTU2.TCNT.

[4] Read TWCRB and then set bit SCC in TWCRB to 1 to suppress synchronous counter clearing for MTU0 to MTU2 and MTU6 and MTU7*1. Here, do not modify the CCE and WRE bit values in TWCRB of MTU6 and MTU7. Synchronous counter clearing for MTU0 to MTU2 and MTU6 and MTU7 is suppressed in the synchronous counter clearing-suppressed periods.

Note 1. The TWCRB.SCC bit value can be modified during count operation. However, if a synchronous clearing occurs when bit TWCRB.SCC is modified from 0 to 1, the synchronous clearing may not be suppressed. If a synchronous clearing occurs when bit TWCRB.SCC is modified from 1 to 0, the synchronous clearing may be suppressed.

Figure 16.73 Example of Procedure for Suppressing Synchronous Counter Clearing for MTU0 to MTU2, and MTU6 and MTU7

- Examples of Suppression of Synchronous Counter Clearing for MTU 0 to MTU2, and MTU6 and MTU7

Figure 16.74 to **Figure 16.77** show examples of operation in which MTU6 and MTU7 operate in complementary PWM mode and synchronous counter clearing for MTU 0 to MTU2, and MTU6 and MTU7 is suppressed by setting the SCC bit in TWCRB in MTU6 and MTU7 to 1. In the examples shown in **Figure 16.74** to **Figure 16.77**, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in **Figure 16.66**, respectively. In these examples, the WRE bit in TWCRB in MTU6 and MTU7 is set to 1.

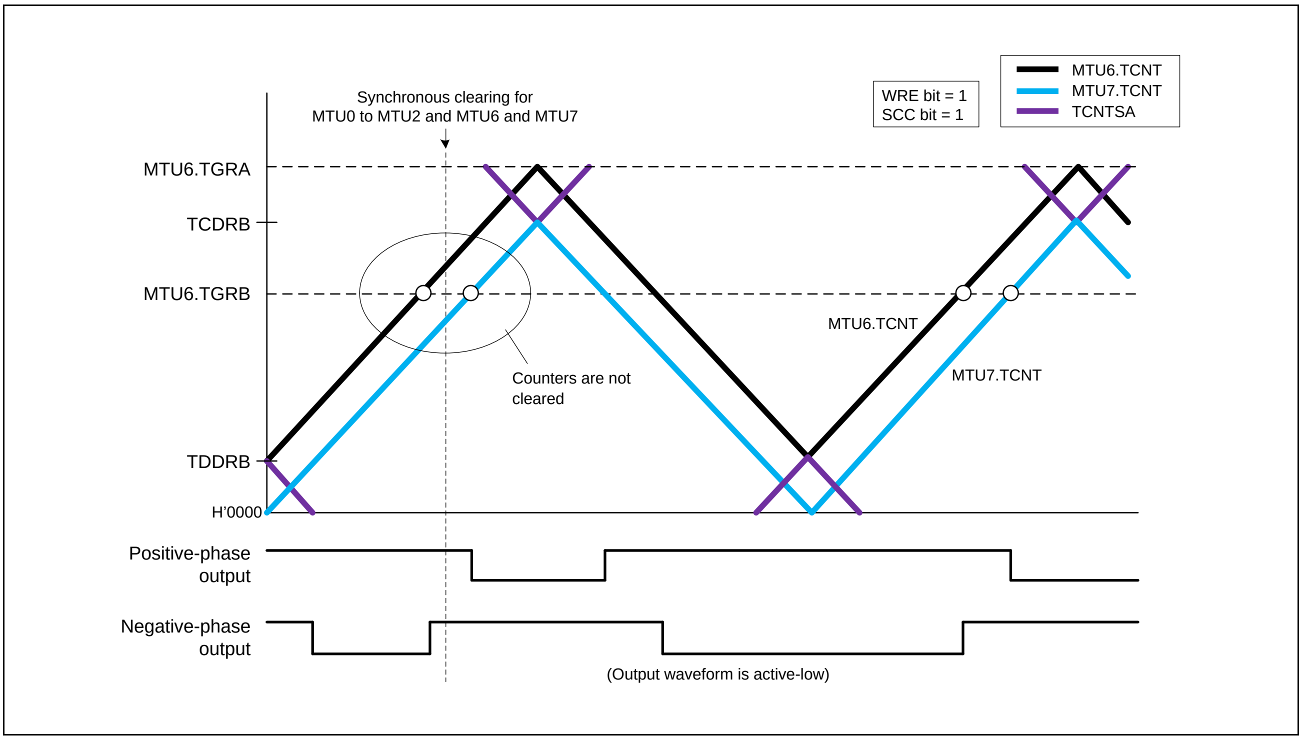


Figure 16.74 Example of Synchronous Clearing in Dead Time during Up-Counting
(Timing (3) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

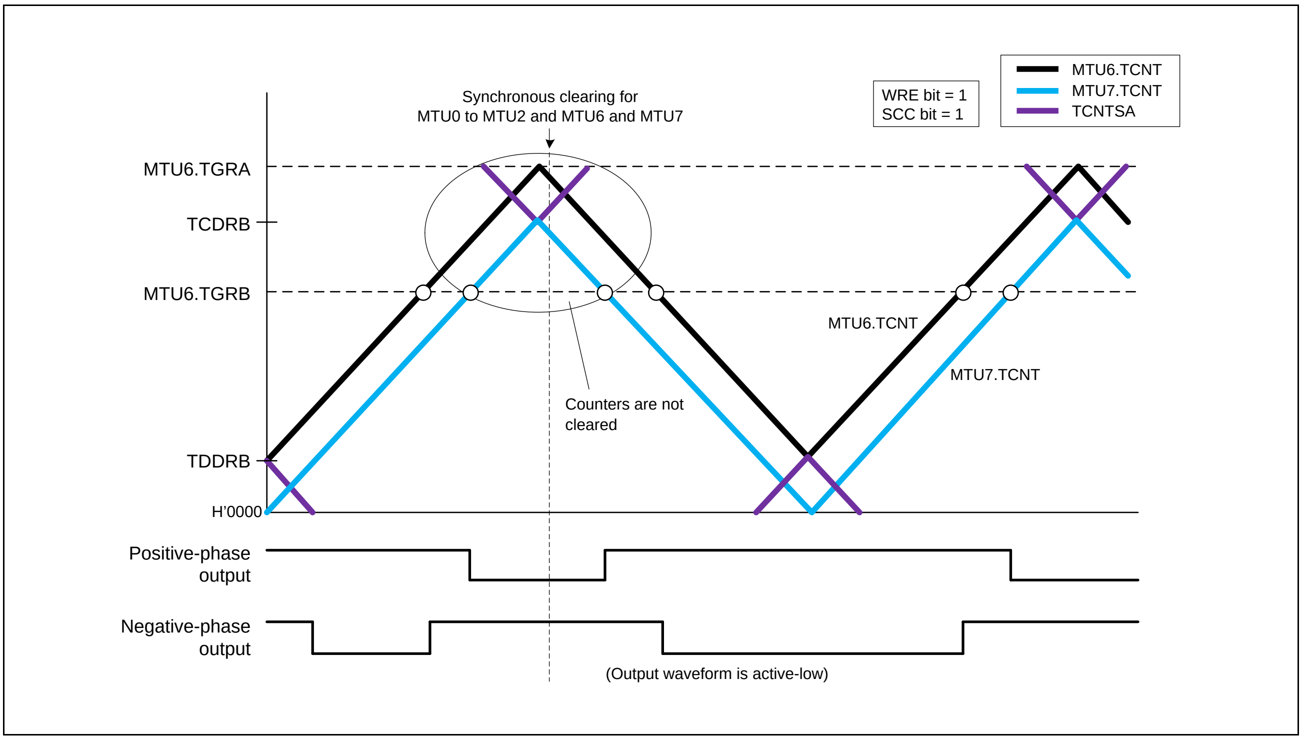


Figure 16.75 Example of Synchronous Clearing in Tb1 interval
(Timing (6) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

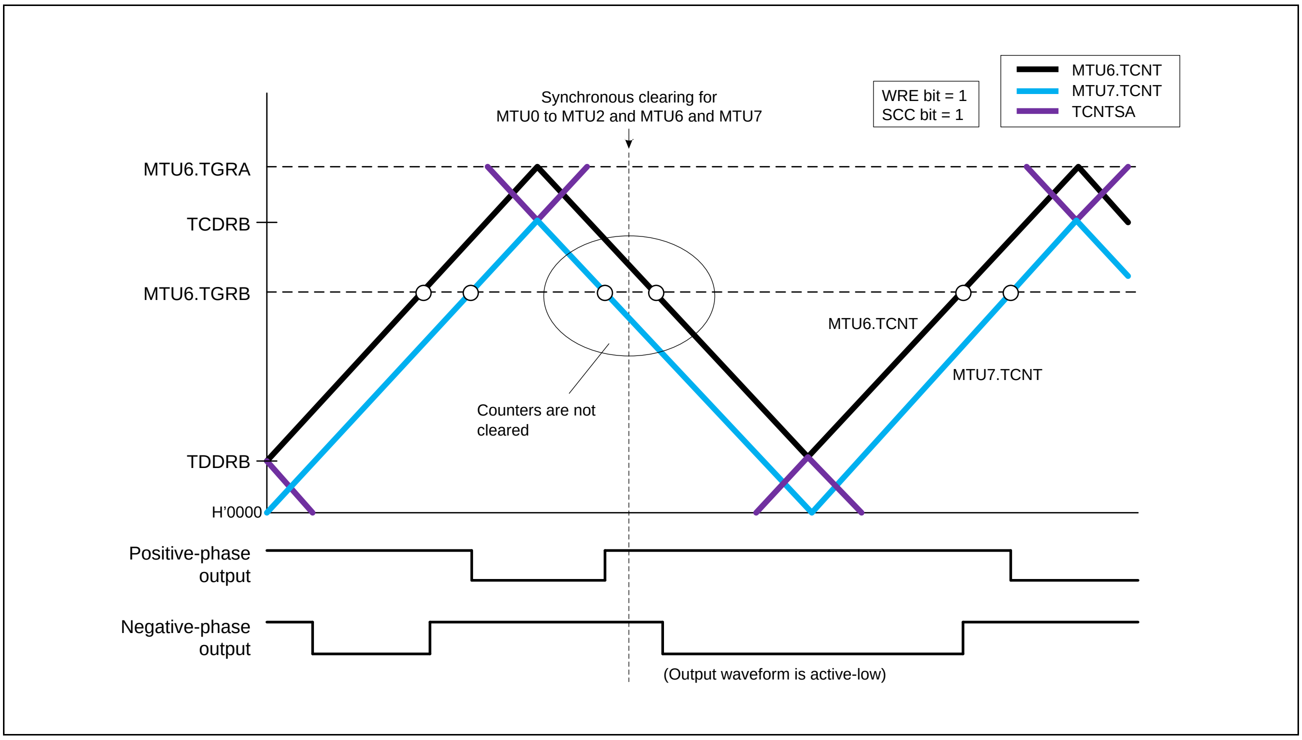


Figure 16.76 Example of Synchronous Clearing in Dead Time during Down-Counting
(Timing (8) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

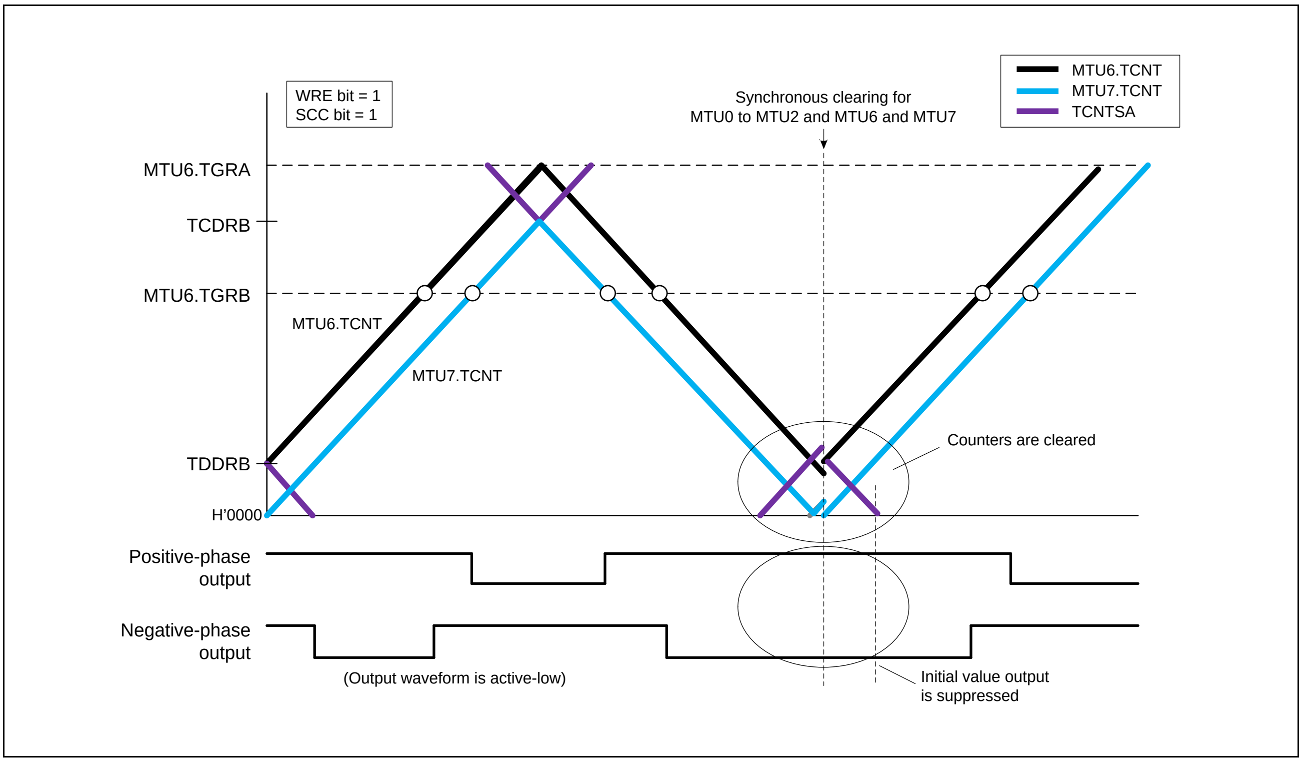


Figure 16.77 Example of Synchronous Clearing in Tb2 interval
(Timing (11) in **Figure 16.66**; TWCRB.WRE Bit is 1 and SCC Bit is 1 in MTU6 and MTU7)

(p) Counter Clearing by MTU3.TGRA (MTU6.TGRA) Compare Match

In complementary PWM mode, MTU3.TCNT, MTU4.TCNT, and TCNTSA (MTU6.TCNT, MTU7.TCNT, and TCNTSB) can be cleared by MTU3.TGRA (MTU6.TGRA) compare match when the TWCRA.CCE (TWCRB.CCE) bit.

Figure 16.78 illustrates an operation example.

NOTES

1. Use this function only in complementary PWM mode 1 (transfer at crest).
2. Do not specify synchronous clearing by another channel (do not set 1 in the SYNC0 to SYNC4 or SYNC6 to SYNC7 bits in the timer synchronous register (TSYRA or TSYRB) and the CEOA to CEOD or CEIA to CEID bits in TSYCR).
3. Do not set the PWM duty value to H’0000.
4. Do not set the PSYE bit in timer output control register 1 (TOCR1A or TOCR1B) to 1.

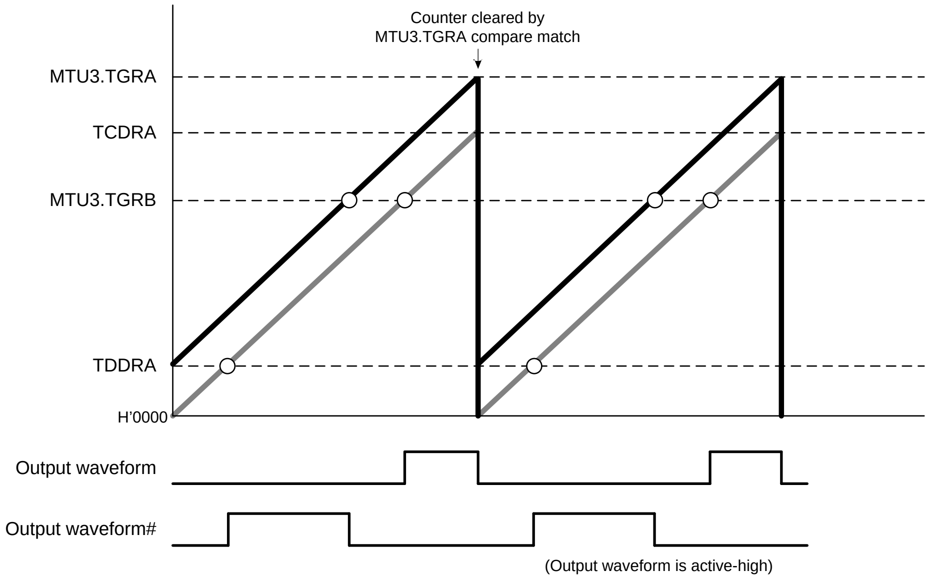


Figure 16.78 Example of Counter Clearing Operation by MTU3.TGRA Compare Match

(q) Example of Waveform Output for Driving AC Synchronous Motor (Brushless DC Motor)

In complementary PWM mode when MTU3 and MTU4 are used, a brushless DC motor can easily be controlled using TGCRA. **Figure 16.79** to **Figure 16.82** show examples of brushless DC motor driving waveforms created using TGCRA.

To switch the output phases for a 3-phase brushless DC motor by means of external signals detected with a Hall element, etc., clear the TGCRA.FB bit to 0. In this case, the external signals indicating the magnetic pole position should be input to pins MTIOC0A, MTIOC0B, and MTIOC0C in MTU0 (make appropriate settings with the port mode registers (PMn) and Pmn port function control register (PFCm) of the GPIO ports). When an edge is detected at pin MTIOC0A, MTIOC0B, or MTIOC0C, the output on/off state is switched automatically.

When the TGCRA.FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in TGCRA is cleared to 0 or set to 1.

The driving waveforms are output from the 6-phase PWM output pins for complementary PWM mode.

With this 6-phase output, while the output is turned on, chopping output is available through complementary PWM mode output function by setting the N bit or P bit in TGCRA to 1. When the N bit or P bit is 0, the level output is selected.

The active level of the 6-phase output (on output level) can be set with the TOCR1A.OLSN and TOCR1A.OLSP bits regardless of the setting of the N and P bits.

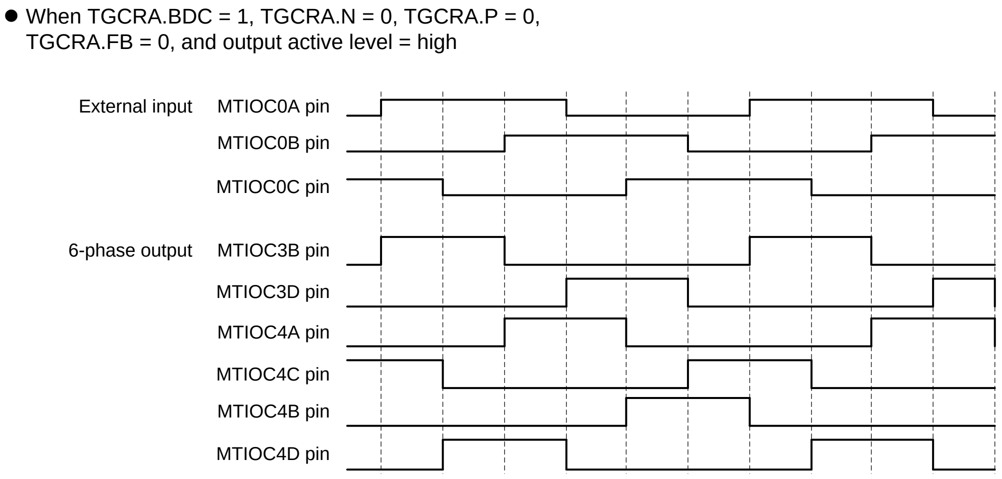


Figure 16.79 Example of Output Phase Switching by External Input (1)

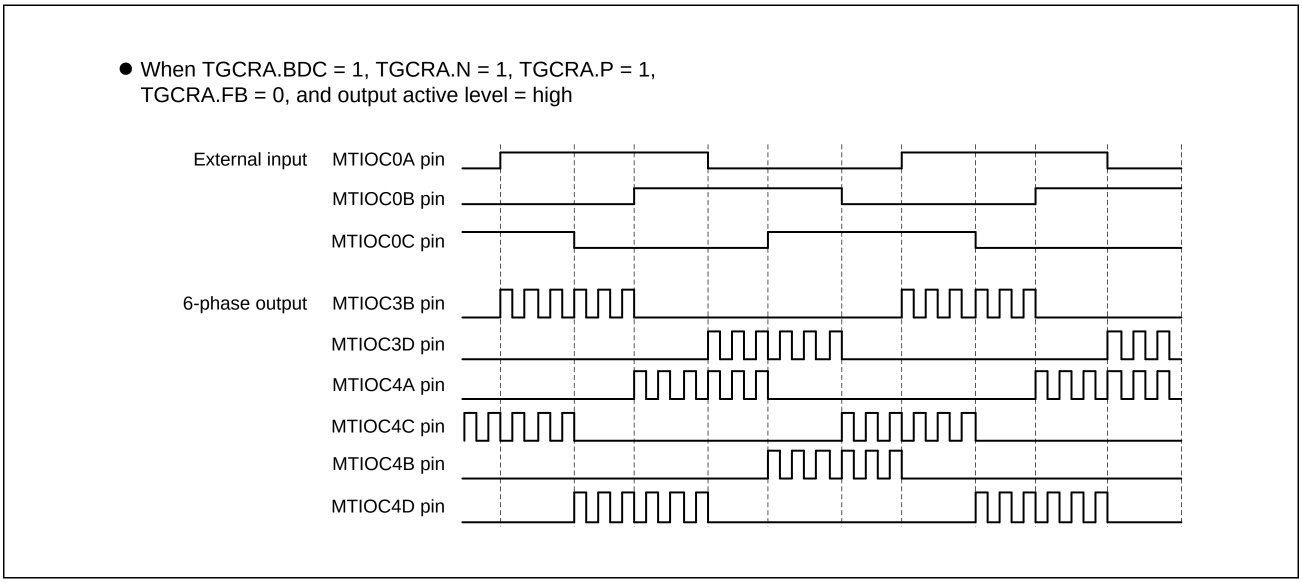


Figure 16.80 Example of Output Phase Switching by External Input (2)

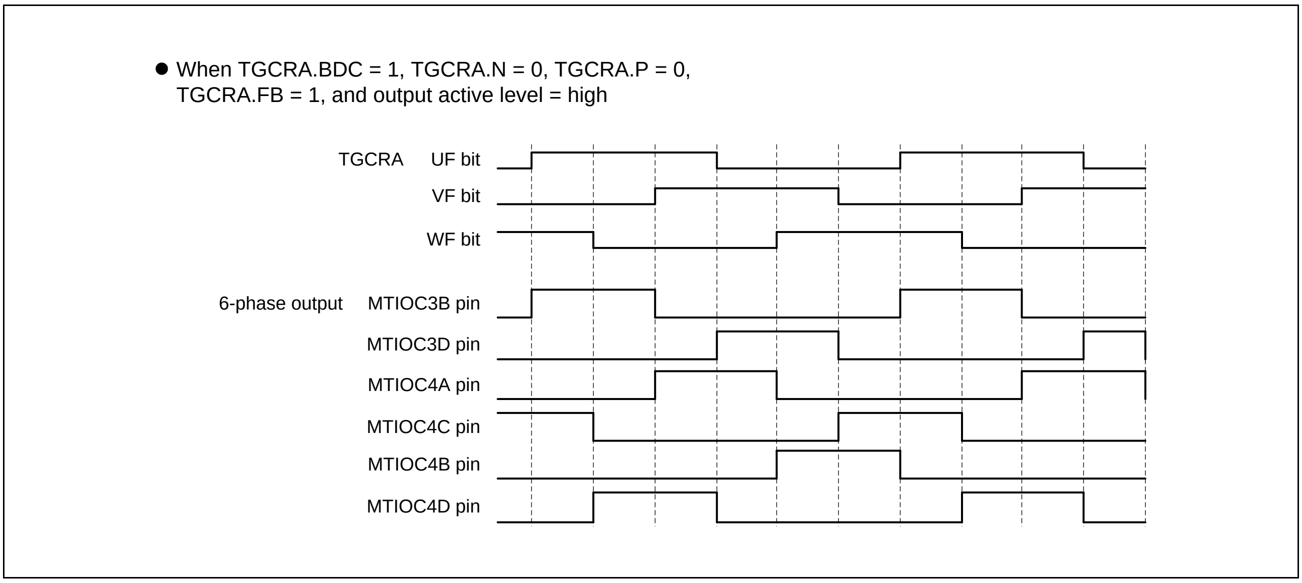


Figure 16.81 Example of Output Phase Switching through UF, VF, and WF Bit Settings (1)

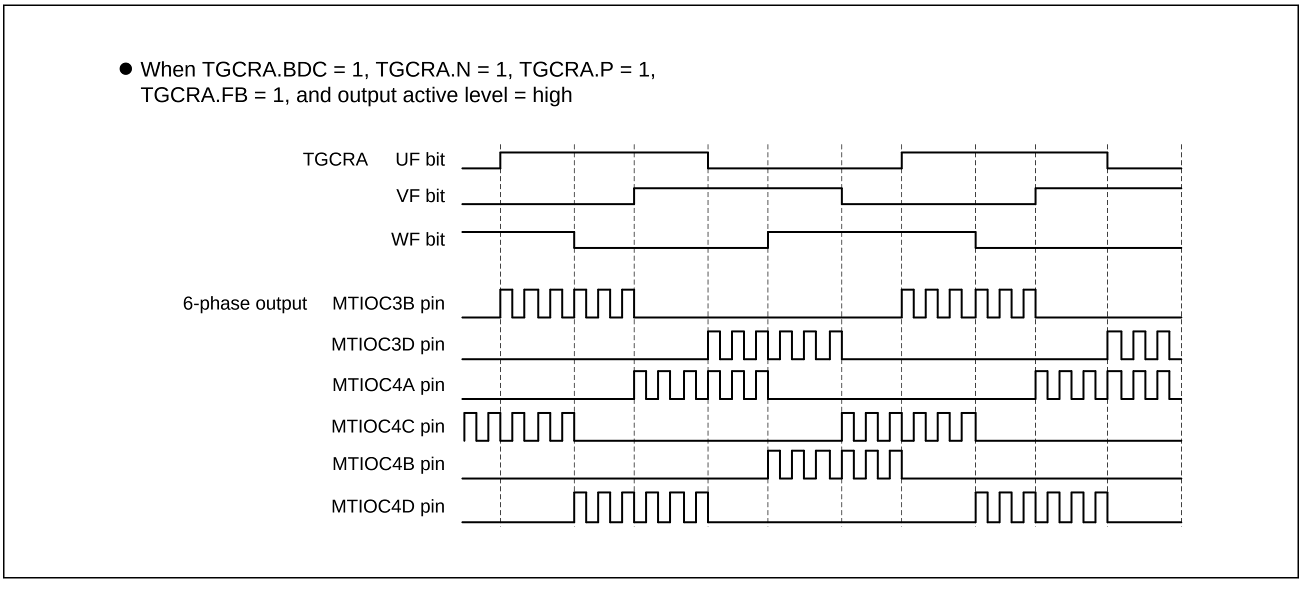


Figure 16.82 Example of Output Phase Switching through UF, VF, and WF Bit Settings (2)

(r) A/D Converter Start Request Setting

In complementary PWM mode, an A/D converter start request can be issued using MTU3.TGRA (MTU6.TGRA) compare match, MTU4.TCNT (MTU7.TCNT) underflow (trough), or compare match on a channel other than MTU3 and MTU4 (MTU6 and MTU7).

When start requests using MTU3.TGRA (MTU6.TGRA) compare match are specified, A/D conversion can be started at the crest of the MTU3.TCNT (MTU6.TCNT) count.

A/D converter start requests can be specified by setting the TIER.TTGE bit. To issue an A/D converter start request at an MTU4.TCNT (MTU7.TCNT) underflow (trough), set the MTU4.TIER.TTGE2 (MTU7.TIER.TTGE2) bit to 1.

(s) Double Buffer Function in Complementary PWM Mode

In complementary PWM mode 3 (transfer at the crest and trough), the PWM output setting resolution can be improved from ±2 to ±1 by setting the TMDR2A.DRS (TMDR2B.DRS) bit to 1.

When setting buffer registers A (TGRD in MTU3, TGRC and TGRD in MTU4, TGRD in MTU6, and TGRC and TGRD in MTU7), set also buffer registers B (TGRE in MTU3, TGRE and TGRF in MTU4, TGRE in MTU6, and TGRE and TGRF in MTU7) at the same time. Each buffer register B should be set to the buffer register A value or (buffer register A value – 1). For details of the setting procedure, refer to **Section 16.3.8(1), Example of Complementary PWM Mode Setting Procedure**.

NOTE

When a buffer register B is set to the buffer register A value, symmetric PWM waveforms are output. When a buffer register B is set to (buffer register A value – 1), asymmetric PWM waveforms are output.

Figure 16.83 shows an example of double buffer operation. Each register data is transferred as follows.

- After TGRD (buffer A) in MTU4 or MTU7 is written to, data is transferred from TGRD (buffer A) in MTU4 or MTU7 to Temp3A or Temp6A (temporary A) and from TGRF (buffer B) in MTU4 or MTU7 to Temp3B or Temp6B (temporary B).
- With timing (1) in the figure, data is transferred from Temp3A or Temp6A (temporary A) to TGRB (compare) in MTU4 or MTU7.
- With timing (2) in the figure, data is transferred from Temp3B or Temp6B (temporary B) to TGRB (compare) in MTU4 or MTU7.

In the crest interval (Tb1 interval), the compare register and temporary register A are valid; in the trough interval (Tb2 interval), the compare register and temporary register B are valid.

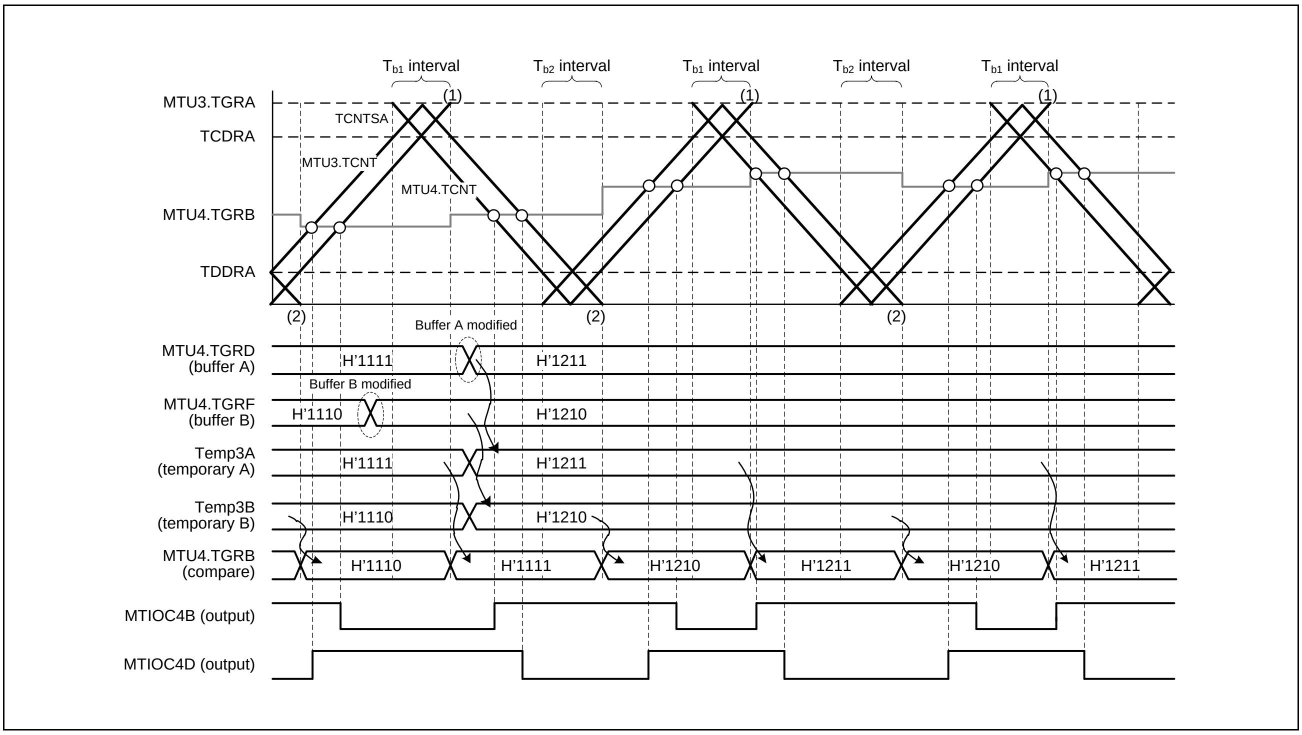


Figure 16.83 Example of Double Buffer Operation

Figure 16.84 shows an example when the buffer write value is smaller than the TDDRA (TDDRB) value, and **Figure 16.85** shows an example when the write value is greater than TCDRA (TCDRB).

In the crest interval, the output is controlled according to the compare match with the compare register or temporary register A; in the trough interval, the output is controlled according to the compare match with the compare register or temporary register B.

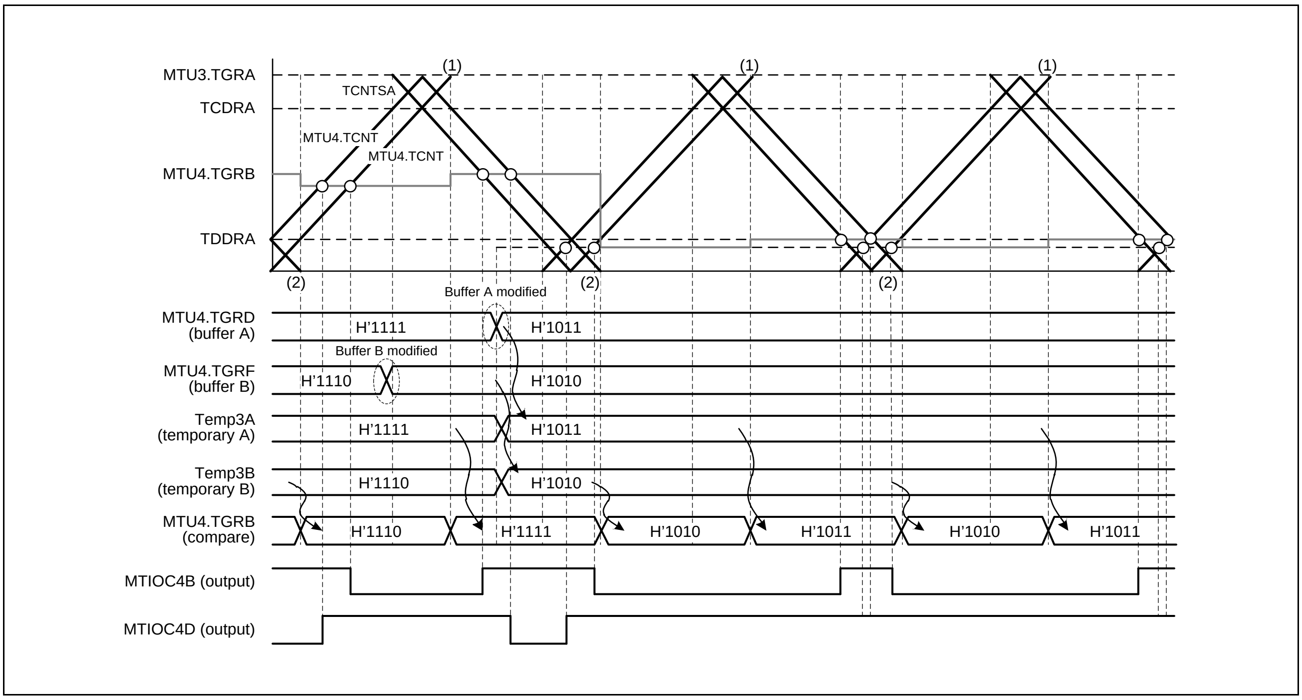


Figure 16.84 Example of Double Buffer Operation (Buffer Write Value is Smaller than TDDRA)

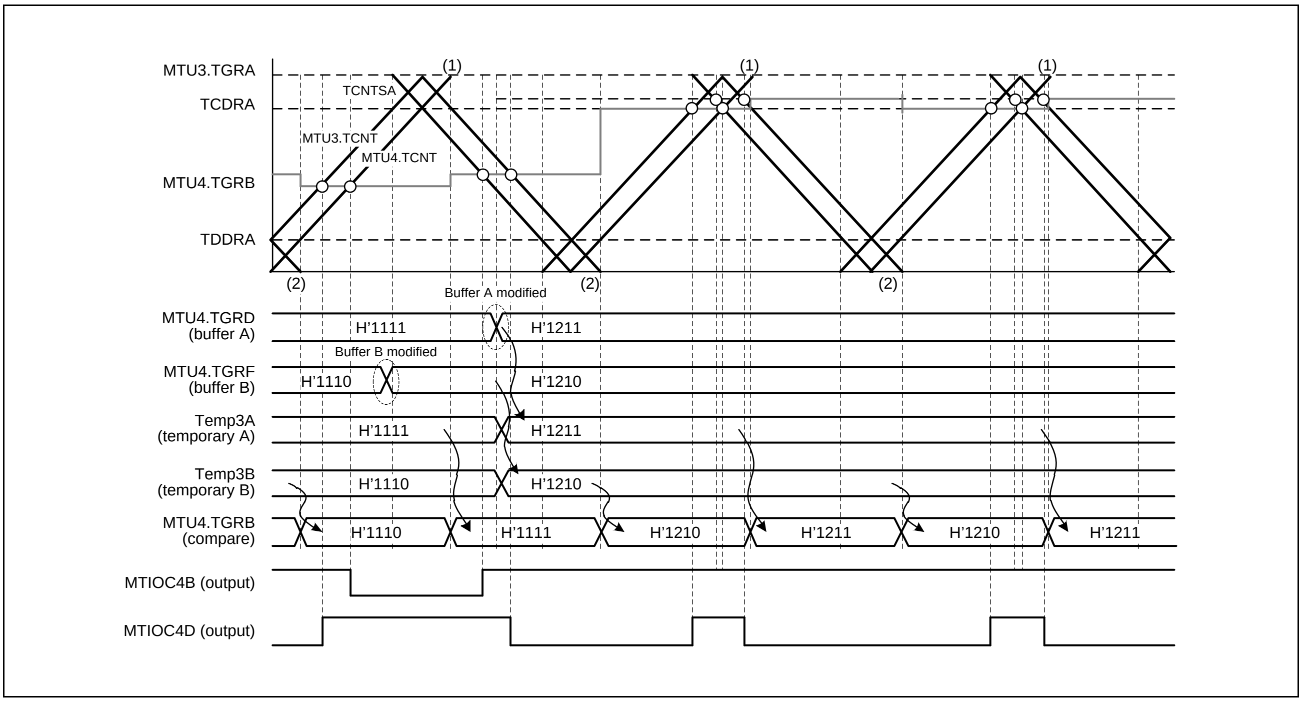


Figure 16.85 Example of Double Buffer Operation (Buffer Write Value is Greater than TCDRA)

(3) Interrupt Skipping Function 1 in Complementary PWM Mode

Interrupts TGIA3 (TGIA6) (at the crest) and TCIV4 (TCIV7) (at the trough) in MTU3 and MTU4 (MTU6 and MTU7) can be skipped up to seven times by making settings in the TITCR1A (TITCR1B) register.

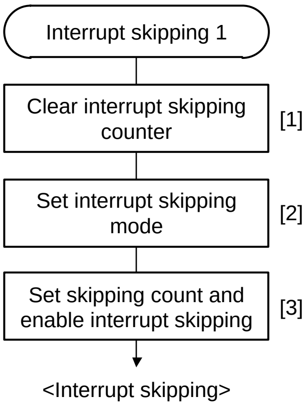
Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in the TBTERA (TBTERB) register. For the linkage with buffer registers, refer to description (c), Buffer Transfer Control Linked with Interrupt Skipping, below.

A/D converter start requests generated by the A/D converter start request delaying function can also be skipped in coordination with interrupt skipping by making settings in the MTU4.TADCR (MTU7.TADCR) register. For the linkage with the A/D converter start request delaying function, refer to **Section 16.3.9, A/D Converter Start Request Delaying Function**.

The TITCR1A (TITCR1B) register should be set while interrupt skipping function 1 is selected by setting the TITM bit in the timer interrupt skipping mode register (TITMRA or TITMRB) to 0, TGIA3 (TGIA6) interrupt requests are disabled by setting the MTU3.TIER (MTU6.TIER) register, TCIV4 (TCIV7) interrupt requests are disabled by setting the MTU4.TIER (MTU7.TIER) register, and a compare match is not generated. Before changing the skipping count, be sure to clear the T3AEN (T6AEN) and T4VEN (T7VEN) bits to 0 to clear the skipping counter.

(a) Example of Interrupt Skipping Function 1 Setting Procedure

Figure 16.86 shows an example of the interrupt skipping function 1 setting procedure. **Figure 16.87** shows the periods during which interrupt skipping count can be changed.



[1] Set bits T3AEN and T4VEN in the timer interrupt skipping set register 1 (TITCR1A) to 0 to clear the skipping counter.

[2] Set bit TITM in the timer interrupt skipping mode register (TITMRA) to 0 to select interrupt skipping function 1.

[3] Specify the interrupt skipping count within the range from 0 to 7 times in bits T3ACOR2 to T3ACOR0 and T4VCOR2 to T4VCOR0 in TITCR1A, and enable interrupt skipping through bits T3AEN and T4VEN.

Note: The setting of TITCR1A must be done while the TGIA3 and TCIV4 interrupt requests are disabled by the settings of MTU3.TIER and MTU4.TIER under the conditions in which a compare match is not generated. Before changing the skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter.

Figure 16.86 Example of Interrupt Skipping Function 1 Setting Procedure

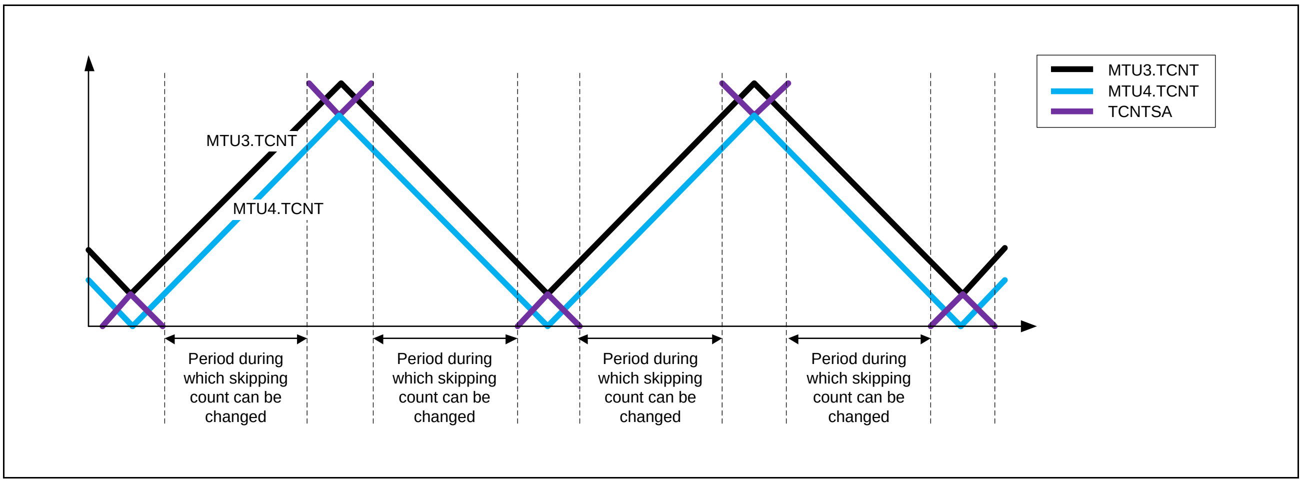


Figure 16.87 Periods during which Interrupt Skipping Count can be Changed

(b) Example of Interrupt Skipping Function 1

Figure 16.88 shows an example of TGIA3 (TGIA6) interrupt skipping in which the interrupt skipping count is set to three by the T3ACOR (T6ACOR) bits and the T3AEN (T6AEN) bit is set to 1 in the TITCR1A (TITCR1B) register.

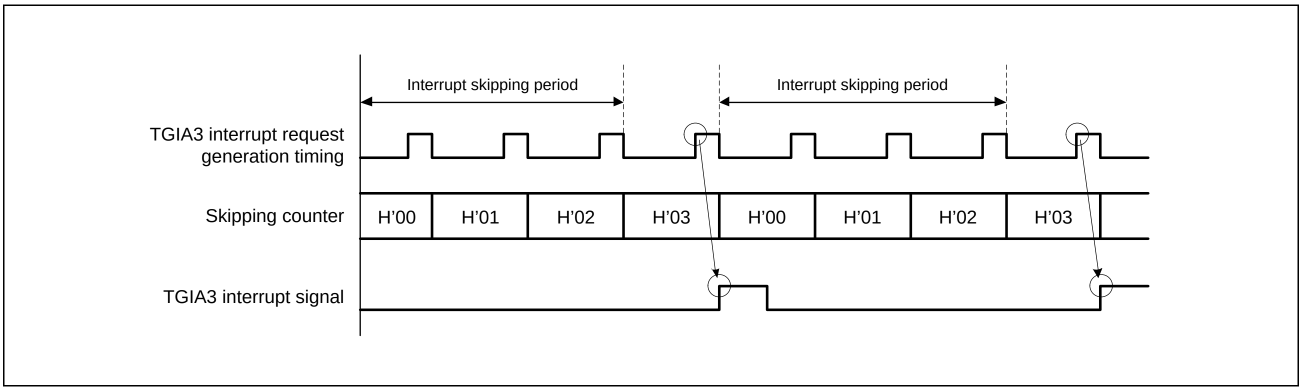


Figure 16.88 Example of Interrupt Skipping Function 1

(c) Buffer Transfer Control Linked with Interrupt Skipping

In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE[1:0] bits in the TBTERA (TBTERB) register.

Figure 16.89 shows an example of operation when buffer transfer is disabled (BTE[1:0] = 01b). While this setting is valid, data is not transferred from the buffer register to the temporary register.

Figure 16.90 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE[1:0] = 10b). While this setting is valid, data is not transferred from the buffer register outside the buffer transfer-enabled period.

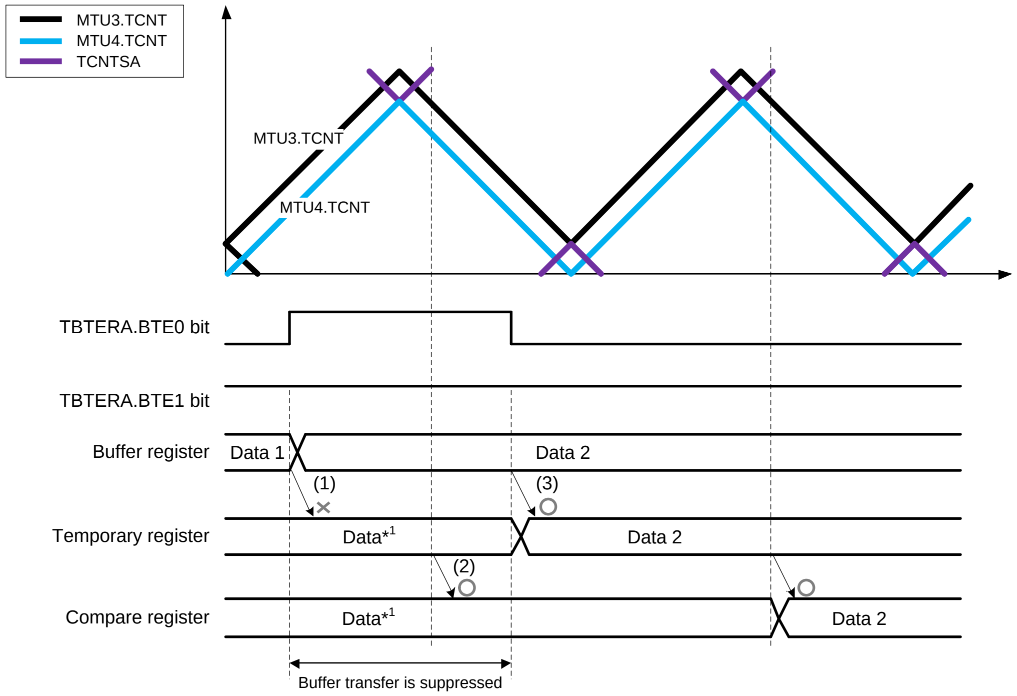
Note that the buffer transfer-enabled period differs depending on whether only the T3AEN (T6AEN) bit in the TITCR1A (TITCR1B) register is set to 1, only the T4VEN (T7VEN) bit in the TITCR1A (TITCR1B) register is set to 1, or both the T3AEN and T4VEN (T6AEN and T7VEN) bits are set to 1. **Figure 16.91** shows the relationship between the T3AEN (T6AEN) and T4VEN (T7VEN) bit settings in TITCR1A (TITCR1B) and buffer transfer-enabled period.

NOTE

This function must be used in combination with interrupt skipping function 1.

When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits in the timer interrupt skipping set register 1 (TITCR1A or TITCR1B) are cleared to 0 or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), make sure that buffer transfer is not linked with interrupt skipping (clear the BTE1 bit in TBTERA or TBTERB to 0).

If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.



(1) No data is transferred from the buffer register to the temporary register in the buffer transfer-disabled period (bits BTE[1:0] in TBTERA are set to 01b).

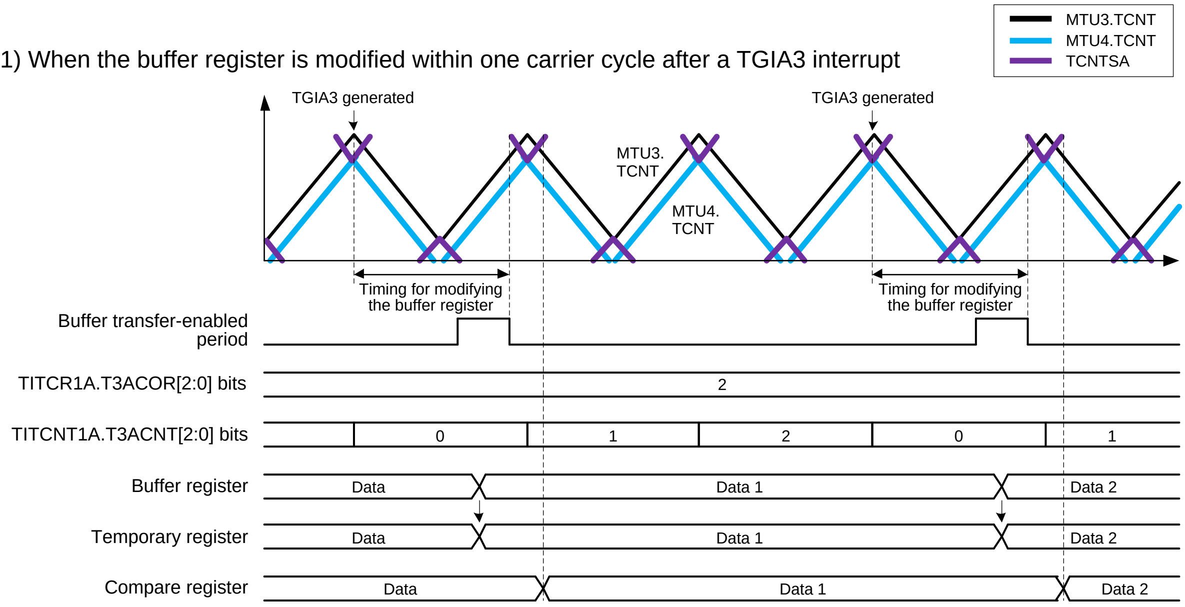
(2) Data is transferred from the temporary register to the compare register even in the buffer transfer-disabled period.

(3) After buffer transfer is enabled, data is transferred from the buffer register to the temporary register.

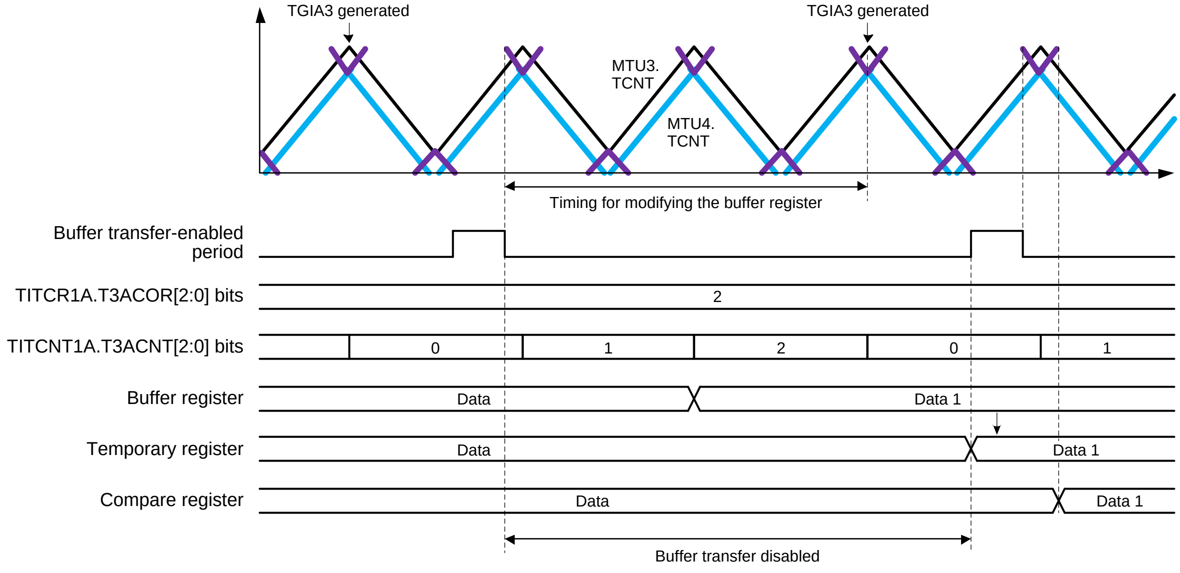
Note 1. When buffer transfer at the crest is selected.

Figure 16.89 Example of Operation When Buffer Transfer is Disabled (BTE[1:0] = 01b)

(1) When the buffer register is modified within one carrier cycle after a TGIA3 interrupt

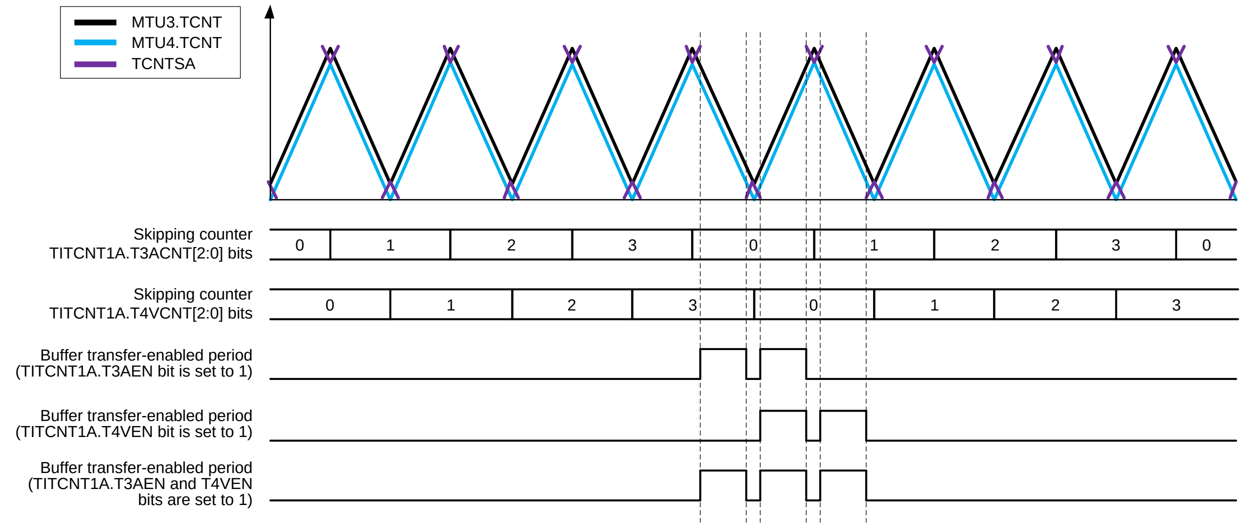


(2) When the buffer register is modified after one carrier cycle has elapsed after a TGIA3 interrupt



Note: Buffer transfer at the crest is selected.
The skipping count is set to two. The TITCR1A.T3AEN bit is set to 1.

Figure 16.90 Example of Operation When Buffer Transfer is Linked with Interrupt Skipping (BTE[1:0] = 10b)



Note: The skipping count is set to three.
Buffer operation is set to transfer at the crest and trough.

Figure 16.91 Relationship between Bits T3AEN and T4VEN in TITCR1A and Buffer Transfer-Enabled Period

(4) Complementary PWM Mode Output Protection Functions

The following output protection functions are provided for complementary PWM mode.

(a) Register and Counter Miswrite Prevention Function

Access from the CPU to the mode registers, control registers, compare registers, and counters can be enabled or disabled by setting the RWE bit in the TRWERA (TRWERB) register. The applicable registers are some of the registers in MTU3, MTU4, MTU6, and MTU7 shown below:

> MTU3.TCR, MTU4.TCR, MTU3.TCR2, MTU4.TCR2, MTU3.TMDR1, MTU4.TMDR1, MTU3.TIORH, MTU4.TIORH, MTU3.TIORL, MTU4.TIORL, MTU3.TIER, MTU4.TIER, MTU3.TCNT, MTU4.TCNT, MTU3.TGRA, MTU4.TGRA, MTU3.TGRB, MTU4.TGRB, MTU.TOERA, MTU.TOCR1A, MTU.TOCR2A, MTU.TGCRA, MTU.TCDRA, MTU.TDDRA MTU6.TCR, MTU7.TCR, MTU6.TCR2, MTU7.TCR2, MTU6.TMDR1, MTU7.TMDR1, MTU6.TIORH, MTU7.TIORH, MTU6.TIORL, MTU7.TIORL, MTU6.TIER, MTU7.TIER, MTU6.TCNT, MTU7.TCNT, MTU6.TGRA, MTU7.TGRA, MTU6.TGRB, MTU7.TGRB, MTU.TOERB, MTU.TOCR1B, MTU.TOCR2B, MTU.TCDRB, and MTU.TDDRB

47 registers in total

This function can disable CPU access to the mode registers, control registers, and counters to prevent miswriting due to CPU runaway. In the access-disabled state, the applicable registers are read as undefined and writing to these registers is ignored.

(b) Halting of PWM Output by External Signal

The PWM output pins of MTU0, MTU3, MTU4, MTU6, and MTU7 can be set to the high-impedance state automatically.

See **Section 17, Port Output Enable 3 (POE3)**, for details.

16.3.9 A/D Converter Start Request Delaying Function

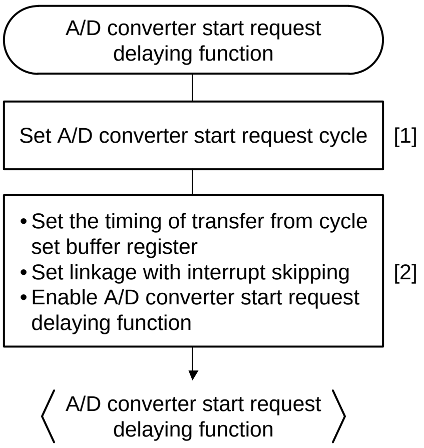
A/D converter start requests can be issued in MTU4 or MTU7 by making settings in the timer A/D converter start request control register (MTU4.TADCR or MTU7.TADCR), timer A/D converter start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB, or MTU7.TADCORA and MTU7.TADCORB), and timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB, or MTU7.TADCOBRA and MTU7.TADCOBRB).

The A/D converter start request delaying function compares MTU4.TCNT with MTU4.TADCORA or MTU4.TADCORB (MTU7.TCNT with MTU7.TADCORA or MTU7.TADCORB), and when their values match, the function issues a respective A/D converter start request (TRG4AN or TRG4BN (TRG7AN or TRG7BN)).

A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be skipped in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the TADCR register in MTU4 (the ITA6AE, ITA7VE, ITB6AE, and ITB7VE bits in the TADCR register in MTU7).

(1) Example of Procedure for Specifying A/D Converter Start Request Delaying Function

Figure 16.92 shows an example of procedure for specifying the A/D converter start request delaying function.



[1] Set the cycle in the timer A/D converter start request cycle buffer register (MTU4.TADCOBRA or MTU4.TADCOBRB) and timer A/D converter start request cycle register (MTU4.TADCORA or MTU4.TADCORB).
(When the timing of transfer from the cycle buffer register to cycle register is set to the trough, or both trough and crest in step [2] below, the same initial value must be specified in the cycle buffer register and cycle register because transfer is performed immediately after counting starts.)

[2] Use bits BF[1:0] in the timer A/D converter start request control register (TADCR in MTU4) to specify the timing of transfer from the timer A/D converter start request cycle buffer register to the A/D converter start request cycle register.

- Specify whether to link with interrupt skipping through bits ITA3AE, ITA4VE, ITB3AE, and ITB4VE.
- Use bits UT4AE, DT4AE, UT4BE, and DT4BE to enable A/D converter start requests (TRG4AN or TRG4BN).

Note 1. Set up TADCR in MTU4 while TCNT in MTU4 is stopped.

Note 2. Set bit BF1 of TADCR in MTU4 to 0 when complementary PWM mode is not selected.

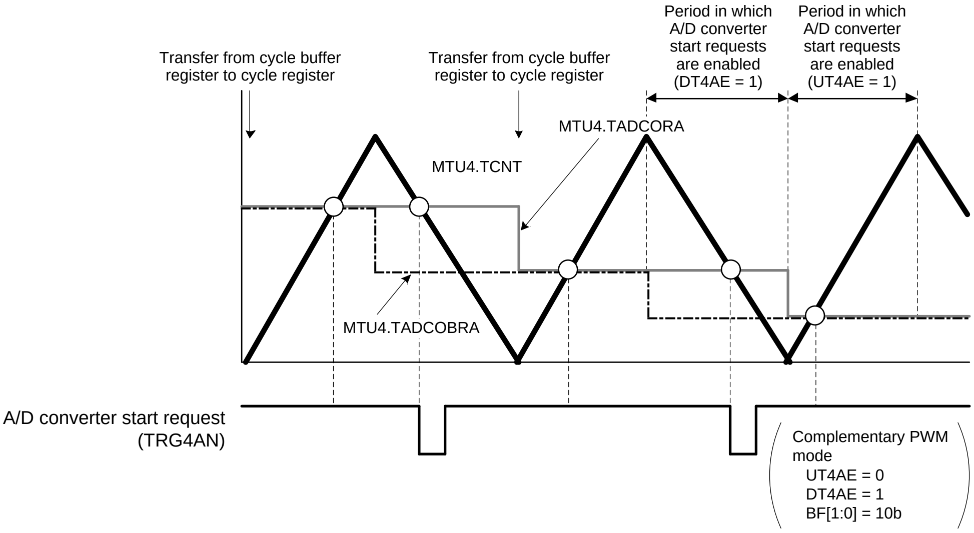
Note 3. Set bits ITA3AE, ITA4VE, ITB3AE, ITB4VE, DT4AE, and DT4BE to 0 when complementary PWM mode is not selected.

Note 4. Set bits ITA3AE, ITA4VE, ITB3AE, and ITB4VE to 0 when interrupt skipping is disabled.

Figure 16.92 Example of Procedure for Specifying A/D Converter Start Request Delaying Function (MTU3 and MTU4)

(2) Basic Example of A/D Converter Start Request Delaying Function Operation

Figure 16.93 shows a basic example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when the trough of MTU4.TCNT (MTU7.TCNT) is specified for the buffer transfer timing and an A/D converter start request signal is output during MTU4.TCNT (MTU7.TCNT) down-counting.



Period in which A/D converter start requests are enabled (UT4AE = 1): 0 ≤ MTU4.TCNT ≤ TCDR – 1
Period in which A/D converter start requests are enabled (DT4AE = 1): TCDR ≥ MTU4.TCNT ≥ 1
UT4AE, DT4AE, BF[1:0]: Bits in TADCR

Figure 16.93 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation

(3) Period in which A/D Converter Start Requests are Enabled

An A/D converter start request (TRG4AN or TRG4BN) can be issued when the TCNT counter value in MTU4 (MTU7) matches the value of the corresponding cycle set register (TADCORA or TADCORB) in MTU4 (MTU7) within the period specified by the UT4AE or UT4BE (UT7AE or UT7BE) bit of TADCR in MTU4 (MTU7).

When the UT4AE or UT4BE (UT7AE or UT7BE) bit is set to 1 in complementary PWM mode, issuing of AD converter start requests is enabled for the period of TCNT up-counting (0 ≤ TCNT ≤ TCDR - 1) in MTU4 (MTU7). When the DT4AE or DT4BE (DT7AE or DT7BE) bit is set to 1, issuing of AD converter start requests is enabled for the period of TCNT down-counting (TCDR ≥ TCNT ≥ 1) in MTU4 (MTU7) (**Figure 16.93**).

(4) Buffer Transfer

The data in the timer A/D converter start request cycle set registers (MTU4.TADCORA and MTU4.TADCORB, or MTU7.TADCORA and MTU7.TADCORB) is updated by writing data to the timer A/D converter start request cycle set buffer registers (MTU4.TADCOBRA and MTU4.TADCOBRB, or MTU7.TADCOBRA and MTU7.TADCOBRB). Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF[1:0] bits in the MTU4.TADCR (MTU7.TADCR) register.

In complementary PWM mode, data is also transferred from the timer A/D converter start request cycle set buffer registers to the timer A/D converter start request cycle set registers when MTU4.TGRD (MTU7.TGRD) register is updated.

When using buffer transfer in complementary PWM mode, pay attention to the buffer transfer timing. For details, refer to **Section 16.6.26, Notes on A/D Converter Start Request Delaying Function in Complementary PWM Mode**. When complementary PWM mode is not selected, be sure to clear the BF1 bit of the TADCR register in MTU4 or MTU7.

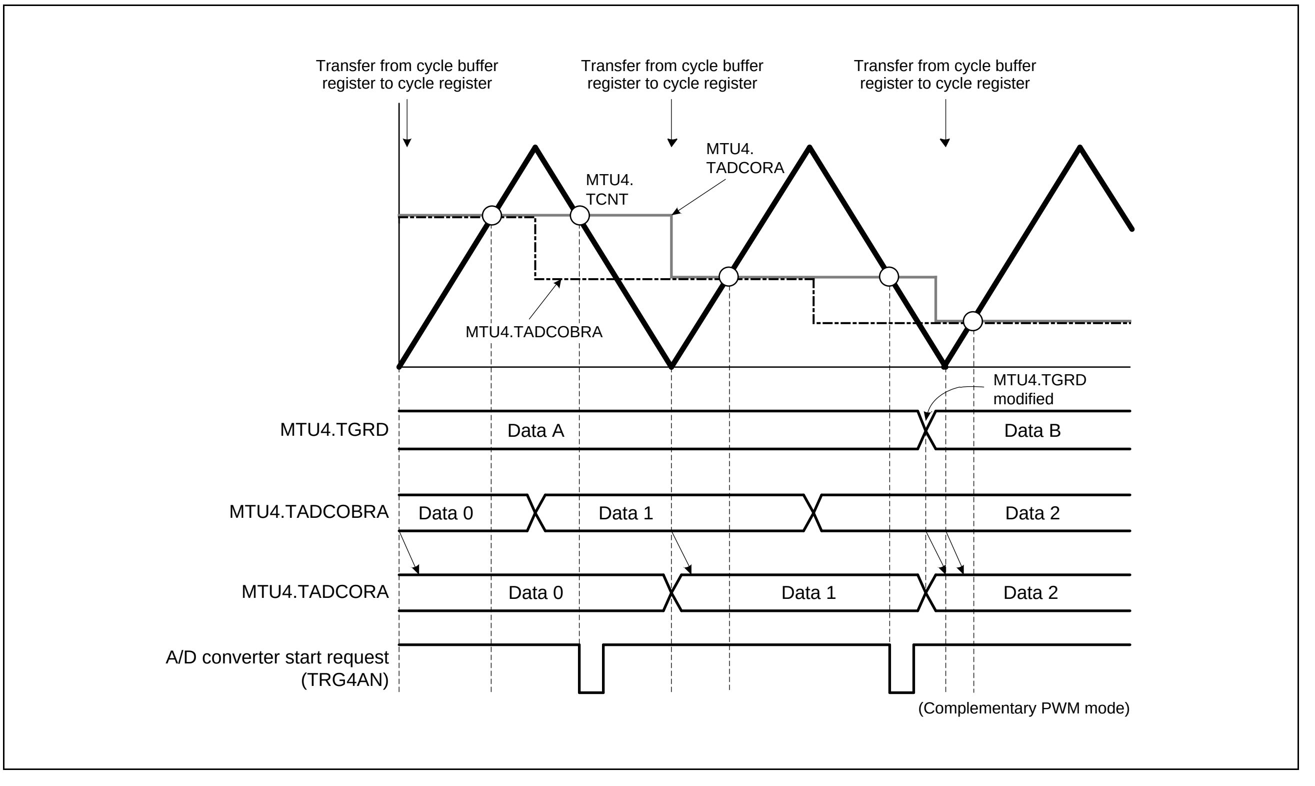


Figure 16.94 Example of A/D Converter Start Request Signal (TRG4AN) and Buffer Transfer Operation

(5) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1

In complementary PWM mode, A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be issued in coordination with interrupt skipping 1 by setting the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits of the TADCR register in MTU4 (MTU7).

Figure 16.95 shows an example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MTU4.TCNT (MTU7.TCNT) up-counting and down-counting and A/D converter start requests are linked with interrupt skipping 1.

Figure 16.96 shows another example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MTU4.TCNT (MTU7.TCNT) up-counting and A/D converter start requests are linked with interrupt skipping 1.

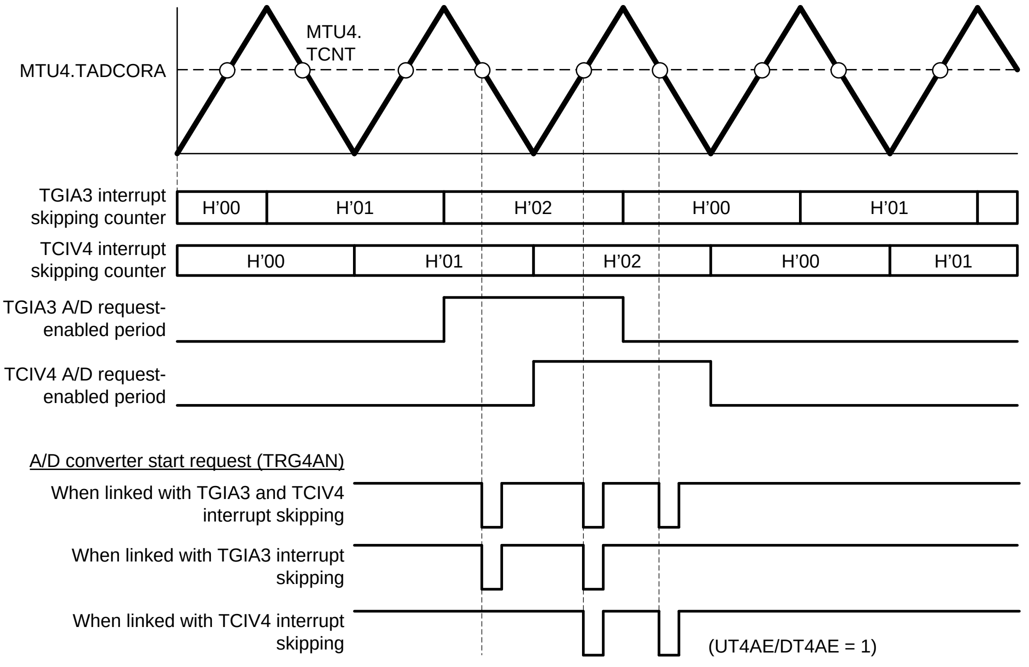
When complementary PWM mode is not selected, A/D converter start request delaying function cannot be used in coordination with interrupt skipping 1; clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits of the TADCR register to 0 in MTU4 (MTU7).

NOTE

This function should be used in combination with interrupt skipping 1.

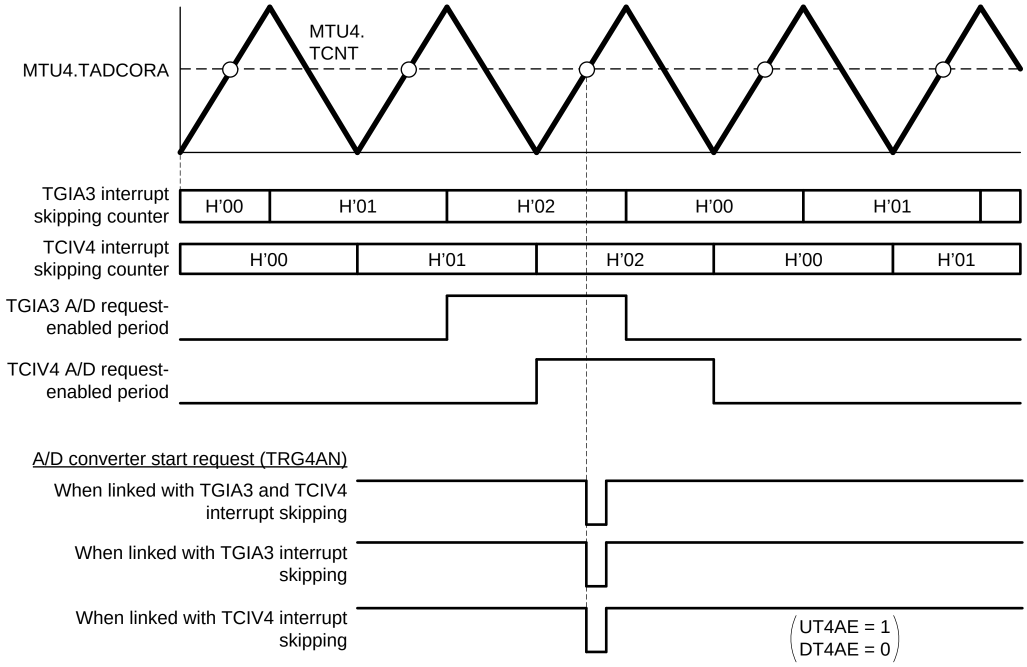
When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits in the timer interrupt skipping set register (TITCR1A (TITCR1B)) are cleared to 0 or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in TITCR1A (TITCR1B) are cleared to 0), make sure that A/D converter start requests are not linked with interrupt skipping 1 (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits in the timer A/D converter start request control register (MTU4.TADCR (MTU7.TADCR)) to 0).

When this function is used, MTU4.TADCORA and MTU4.TADCORB (MTU7.TADCORA and MTU7.TADCORB) should be set with the value ranging H’0002 to the value set in TCDRA minus 2 (value set in TCDRB minus 2).



Note: When the interrupt skipping count is set to two.

Figure 16.95 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE and DT4AE = 1)



Note: When the interrupt skipping count is set to two.

Figure 16.96 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1 (UT4AE = 1, DT4AE = 0)

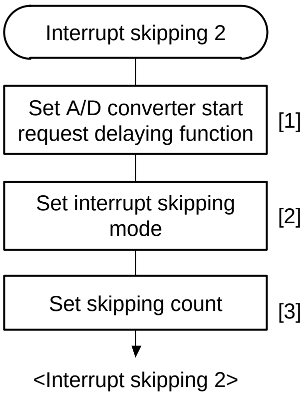
(6) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 2

By setting the TITM bit to 1 in the TITMRA (TITMRB) register, the counter starts down-counting from the value (0 to 7) set in the TRG4COR[2:0] (TRG7COR[2:0]) bits in TITCR2A (TITCR2B) register every time an A/D converter start trigger (TRG4AN or TRG4BN (TRG7AN or TRG7BN)) is generated. When the counter value reaches 0 and is reloaded, the TRG4AN and TRG4BN (TRG7AN and TRG7BN) interrupts become valid and an A/D converter start request signal (TRG4ABN (TRG7ABN)) is output.

This function is valid only when the A/D converter request delaying function is enabled.

(a) Example of Procedure for Setting Interrupt Skipping Function 2

Figure 16.97 shows an example of procedure for setting interrupt skipping function 2.



[1] Set the A/D converter start request delaying function. For details, refer to **Section 16.3.9(1), Example of Procedure for Specifying A/D Converter Start Request Delaying Function**.

[2] Set bit TITM in the TITMRA (TITMRB) register to 1 to select interrupt skipping function 2.

[3] Specify the interrupt skipping count within the range from 0 to 7 times in bits TRG4COR[2:0] in TITCR2A (TITCR2B) register.

Figure 16.97 Example of Procedure for Setting Interrupt Skipping Function 2

(b) Example of Interrupt Skipping Function 2 Operation

Figure 16.98 shows an example of interrupt skipping 2 operation.

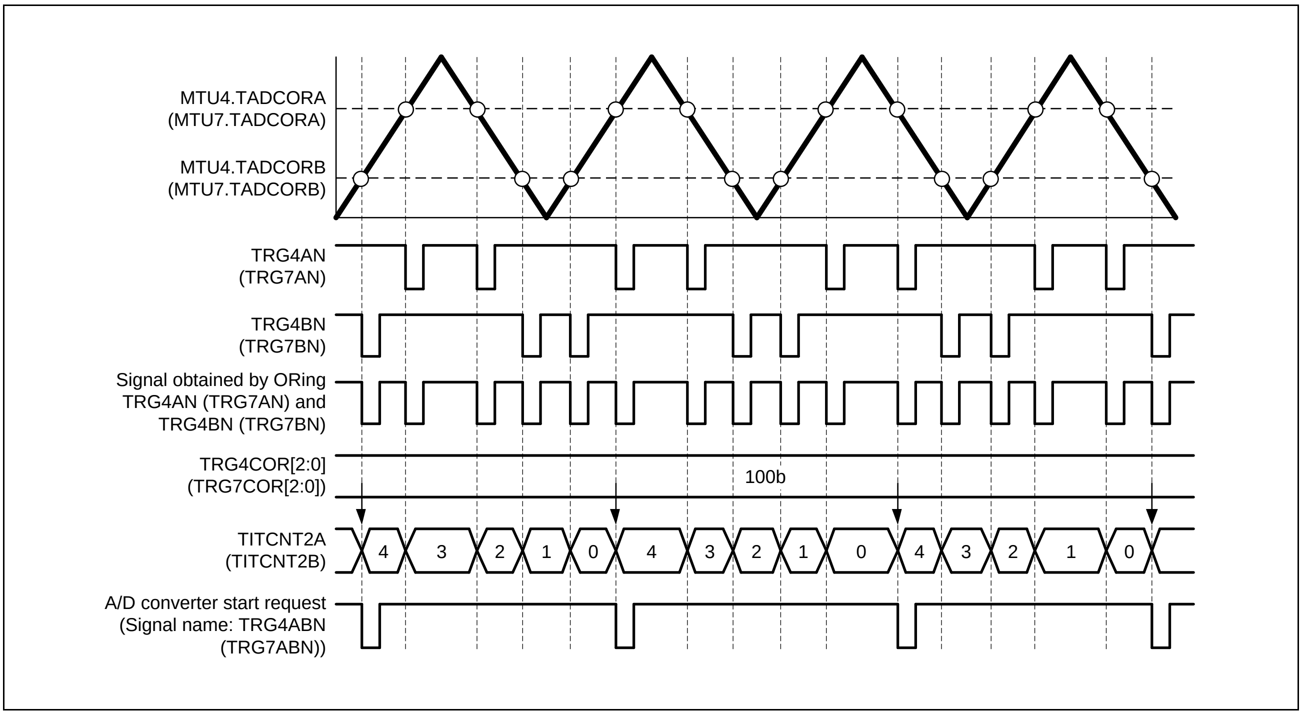


Figure 16.98 Example of Interrupt Skipping 2 Operation (Skipping Count is Set to Four)

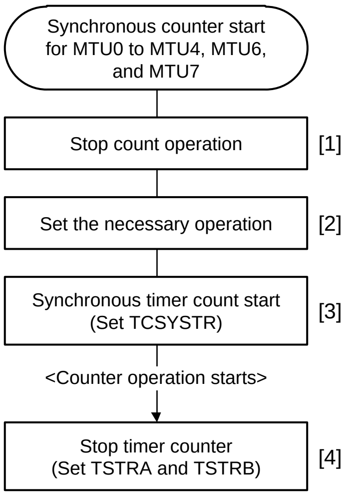
16.3.10 Synchronous Operation of MTU0 to MTU4, MTU6, and MTU7

(1) Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7

The counters in MTU0 to MTU4, MTU6, and MTU7 can be started synchronously by making the TCSYSTR settings.

(a) Example of Procedure for Setting Synchronous Counter Start for MTU0 to MTU4, MTU6, and MTU7

Figure 16.99 shows an example of the procedure for specifying synchronous counter start in MTU0 to MTU4, MTU6, and MTU7.



[1] Set TSTRA and TSTRB to stop the counters used for synchronous start operation.

[2] Specify the necessary operation with appropriate registers such as TCR, TMDR1, and TMDR2.

[3] In TCSYSTR, set the bits corresponding to the counters to be started synchronously to 1 at the same time. When TCSYSTR is set, TSTRA and TSTRB for the target timer counters are automatically set appropriately, the counters start synchronously, and TCSYSTR is automatically cleared.

[4] To stop a timer counter, clear the bit in TSTRA or TSTRB that corresponds to the counter to 0.

Note 1. Clearing a bit in TCSYSTR corresponding to an operating counter to 0 does not stop the counter.

Note 2. To start counters synchronously in reset-synchronized PWM mode or complementary PWM mode using MTU3 and MTU4 or MTU6 and MTU7, set the bits in TCSYSTR that correspond to MTU 3 to MTU4, MTU6, and MTU7 to 1 at the same time.

Figure 16.99 Example of Procedure for Specifying Synchronous Counter Start in MTU0 to MTU4, MTU6, and MTU7

(b) Examples of Synchronous Counter Start Operation

Figure 16.100 shows an example of the synchronous counter start operation in MTU0 to MTU4, MTU6, and MTU7.

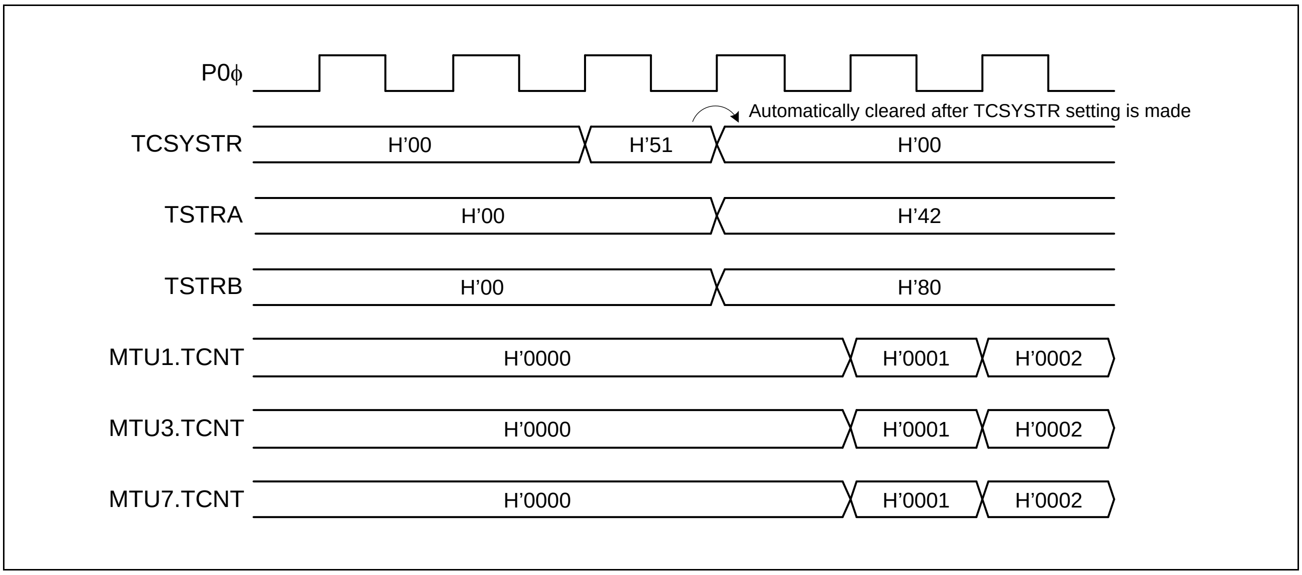


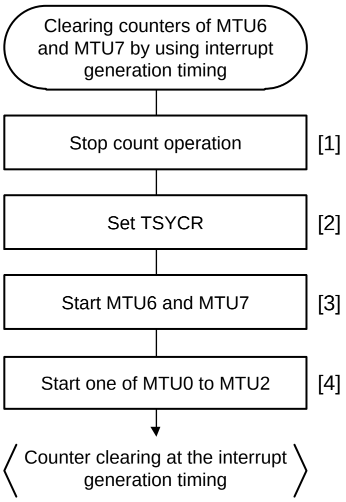
Figure 16.100 Example of Synchronous Counter Start Operation in MTU0 to MTU4, MTU6, and MTU7

(2) Synchronous Counter Clearing for MTU6 and MTU7

The counters in MTU6 and MTU7 can be cleared by the TGImn interrupt generation timing (m = A to D; n = 0 to 2) through the TSYCR setting.

(a) Example of Procedure for Specifying Synchronous Counter Clearing for MTU6 and MTU7

Figure 16.101 shows an example of the procedure for specifying synchronous counter clearing in MTU6 and MTU7 by using the interrupt generation timing.



[1] Set TSTRA and TSTRB to stop the counters used for this function.

[2] Use TSYCR to specify the source to be used for the MTU6.TCNT and MTU7.TCN counter clearing.

[3] Start MTU6.TCNT or MTU7.TCNT.

[4] Start MTU0.TCNT, MTU1.TCNT, or MTU2.TCNT.

Note: The TSYCR setting is ignored while the counter is stopped. The setting becomes valid after MTU6.TCNT or MTU7.TCNT is started.

Figure 16.101 Example of Procedure for Specifying Synchronous Counter Clearing in MTU6 and MTU7

(b) Examples of Synchronous Counter Clearing in MTU6 and MTU7

Figure 16.102 and **Figure 16.103** show examples of synchronous counter clearing in MTU6 and MTU7 by using the interrupt generation timing.

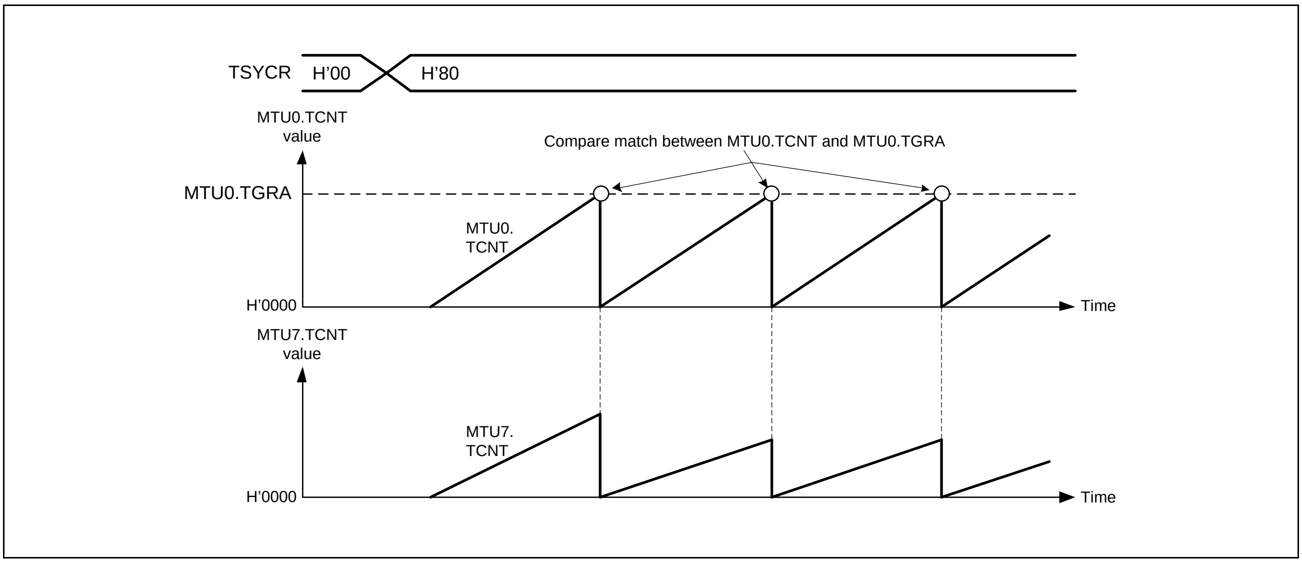


Figure 16.102 Example of Synchronous Counter Clearing in MTU6 and MTU7 (1)

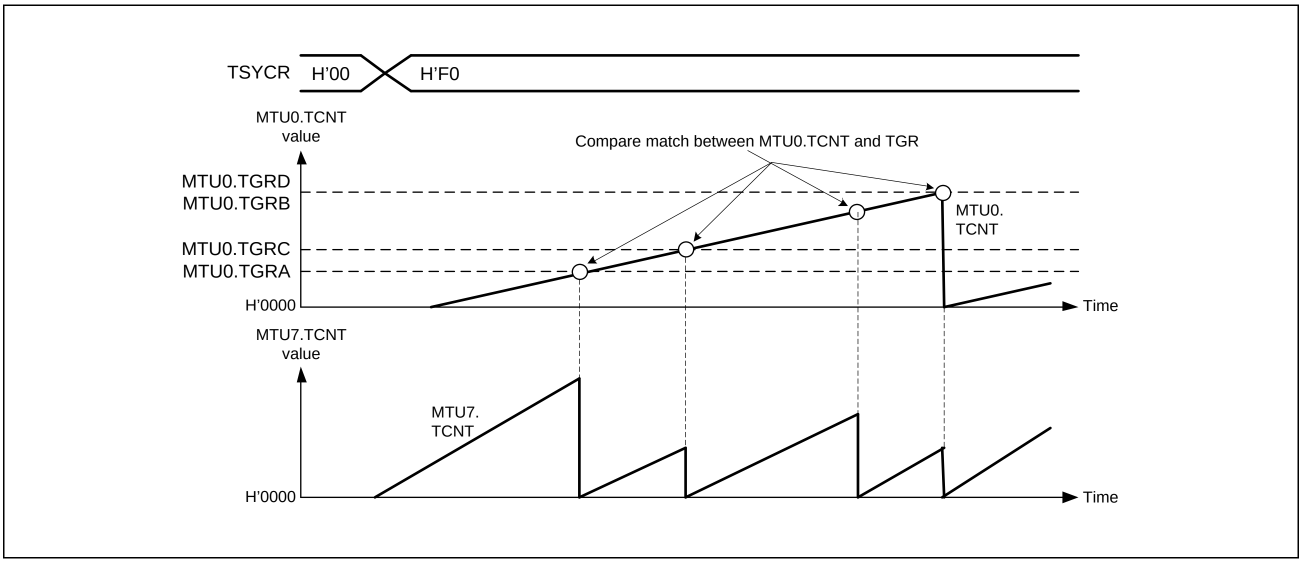


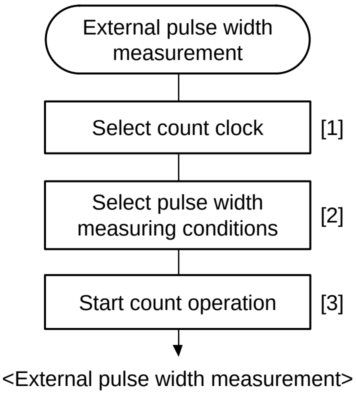
Figure 16.103 Example of Synchronous Counter Clearing in MTU6 and MTU7 (2)

16.3.11 External Pulse Width Measurement

The pulse widths of up to three external input lines can be measured in MTU5.

When the IOC[4:0] bits in MTU5.TIORU, MTU5.TIORV, MTU5.TIORW are set for pulse width measurement, the pulse width of the signal input to the MTIC5U, MTIC5V, and MTIC5W pins are measured. TCNTU, TCNTV, and TCNTW count up while the level specified by the IOC[4:0] bits is input.

Figure 16.104 shows an example of setting external pulse width measurement, and **Figure 16.105** an example of external pulse width measurement.



[1] Use the TPSC1 and TPSC0 bits in MTU5.TCRU, MTU5.TCRV, or MTU5.TCRW to select the count clock.

[2] In MTU5.TIORU, MTU5.TIORV, or MTU5.TIORW, select the high level or low level for the pulse width measuring condition.

[3] Set bit CSTU5, CSTV5, or CSTW5 in MTU5.TSTR to 1 to start count operation.

Note: When MTU6 and MTU7 are not operating in complementary PWM mode, note the following 1 to 3.

1. Set the CMPCLR5U, CMPCLR5V, and CMPCLR5W bits in the TCNTCMPCLR register to 0.
2. Set the TGIE5U, TGIE5V, and TGIE5W bits in MTU5.TIER to 0.
3. The TCNT value is not captured in the TGR register.

Figure 16.104 Example of External Pulse Width Measurement Setting Procedure

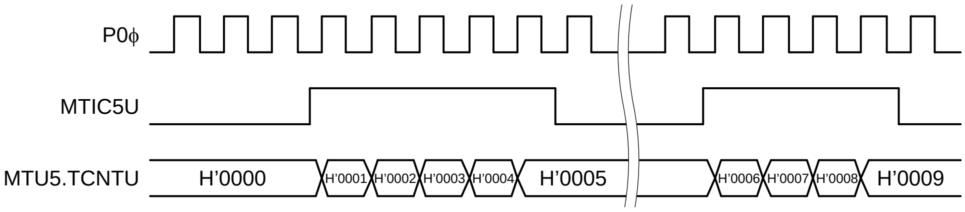


Figure 16.105 Example of External Pulse Width Measurement (Measuring High Pulse Width)

16.3.12 Dead Time Compensation

MTU5 to MTU7 can be used in combination to compensate for the delay in the dead time (the delay between complementary PWM output and inverter output).

Figure 16.106 shows an example of a motor control circuit in which the combination of MTU5 to MTU7 is used to compensate for the dead time delay.

The external pulse measurement function of MTU5 allows the specification of a correction to the duty cycle specified in the PWM output compare registers by measuring the delay between the complementary PWM output and inverter output. This can be used for dead time compensation for the PWM output waveform during complementary PWM operation of MTU6 and MTU7 (**Figure 16.107**).

Figure 16.108 shows the procedure for setting dead time compensation using MTU5 to MTU7. For details on MTU5 operation at this time, refer to **Section 16.3.13, TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode**.

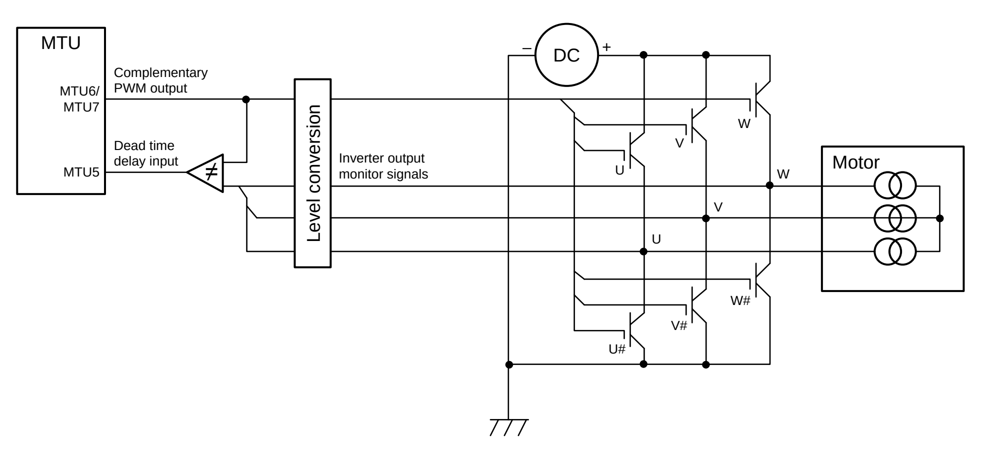


Figure 16.106 Motor Control Circuit Example

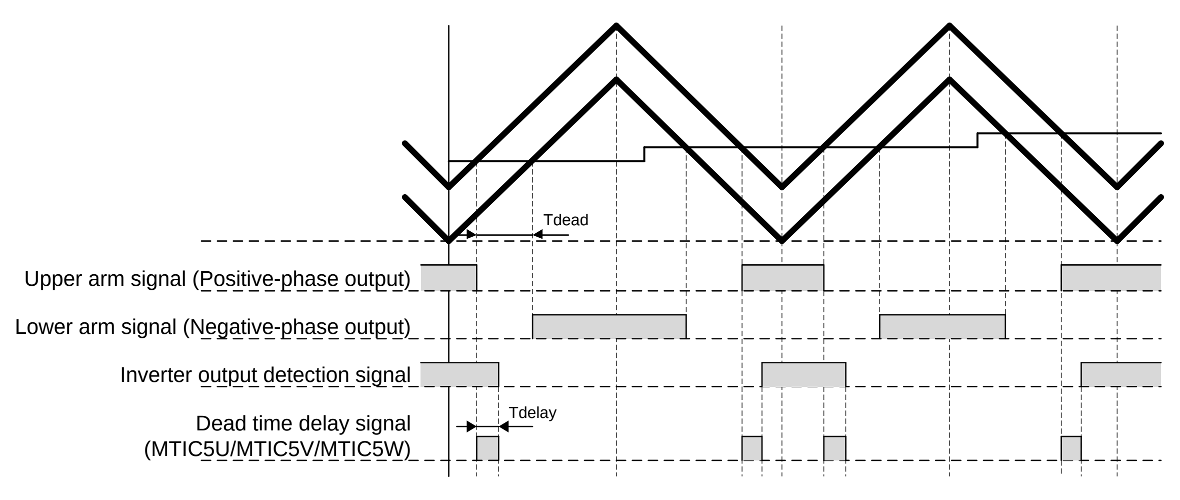
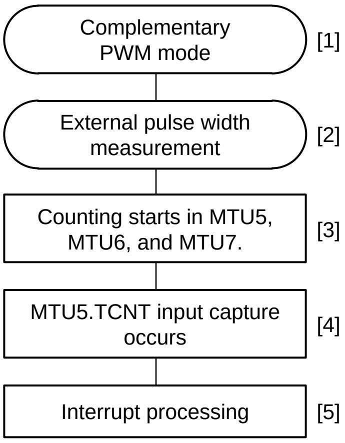


Figure 16.107 Delay in Dead Time in Complementary PWM Operation

(1) Example of Dead Time Compensation Setting Procedure

Figure 16.108 shows an example of dead time compensation setting procedure by using three counters in MTU5.



[1] Set MTU6 and MTU7 to complementary PWM mode. For details, refer to **Section 16.3.8, Complementary PWM Mode**.

[2] Specify the external pulse width measurement function for the target TIOR in MTU5. For details, refer to **Section 16.3.11, External Pulse Width Measurement**.

[3] Set bits CST6 and CST7 in TSTRB and bits CSTU5, CSTV5, and CSTW5 in MTU5.TSTR to 1 to start count operation.

[4] When the capture condition specified in TIOR is satisfied, the MTU5.TCNT value is captured in MTU5.TGR.
To prevent the generation of the MTU5.TGIm5 interrupt (m = U, V, W) request under the capture condition specified in TIOR, do not enable interrupt requests in MTU5.TIER.

[5] For U-phase dead time compensation, when an interrupt is generated at the crest (TGIA6) or trough (TCIV7) in complementary PWM mode, read the MTU5.TGRU value, calculate the difference in time between MTU6.TGRB and MTU5.TGRU, and write the corrected value to MTU6.TGRD in the interrupt processing.
For the V phase and W phase, read the MTU5.TGRV and MTU5.TGRW values and write the corrected values to MTU7.TGRC and TGRD, respectively, in the same way as for U-phase compensation.
The MTU5.TCNT value should be cleared through the TCNTCMPCLR setting or by software.

Note: The port mode registers (PMn) and Pmn port function control register (PFCm) of the GPIO ports should be set up in advance.
As the interrupt flag is set under the capture condition specified in TIOR, do not enable interrupt requests in MTU5.TIER.

Figure 16.108 Example of Dead Time Compensation Setting Procedure

16.3.13 TCNTU, TCNTV, and TCNTW Capture at Crest and/or Trough in Complementary PWM Mode

The external pulse width measurement function of MTU5 can be used to transfer the values of TCNTU, TCNTV, and TCNTW to TGRU, TGRV, and TGRW at the crests, troughs, or both crests and troughs of the complementary PWM output when MTU6 and MTU7 are being used in complementary PWM mode. The type of transfer timing is specified in TIORU, TIORV, and TIORW. When the CMPCLR5U, CMPCLR5V, and CMPCLR5W bits in the TCNTCNPCLR register are set to 1, TCNTU, TCNTV, and TCNTW are cleared to H’0000 at the timing for transfer to TGRU, TGRV, and TGRW.

Note that the TCNTU, TCNTV, and TCNTW capture operation at the crest and/or trough of complementary PWM output in MTU5 is not available when MTU3 and MTU4 are used in complementary PWM mode.

Figure 16.109 shows an operation example in which TCNTU is used as a free-running counter without being cleared and the value is captured in TGRU at the crest and trough in complementary PWM mode.

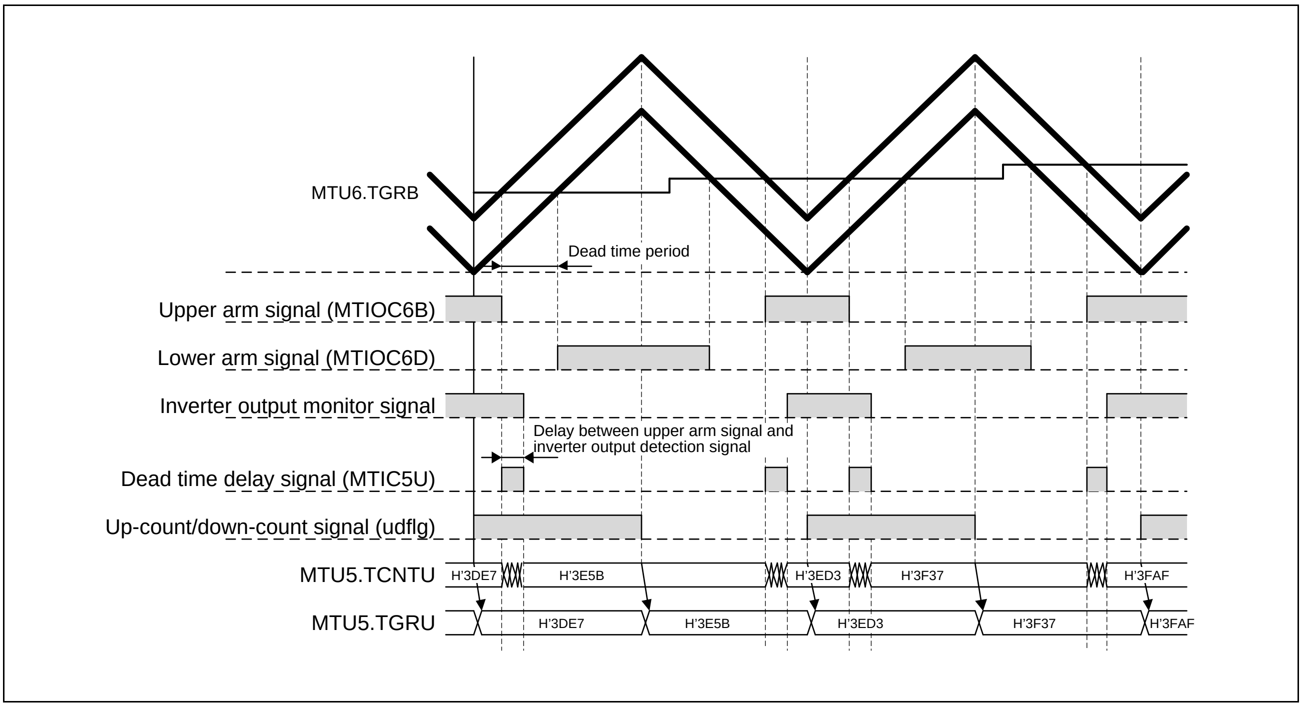


Figure 16.109 TCNTU Capture at Crest and Trough in Complementary PWM Operation

16.3.14 Noise Filter Function

The input capture input pins and external pulse input pins have a noise filter function.

Set the NFCRn register (n = 0 to 7, C) to enable or disable the noise filter function and set the sampling clock. The noise filter for each pin can be enabled or disabled individually, and the sampling clock can be set for each channel.

Figure 16.110 shows the timing of noise filtering.

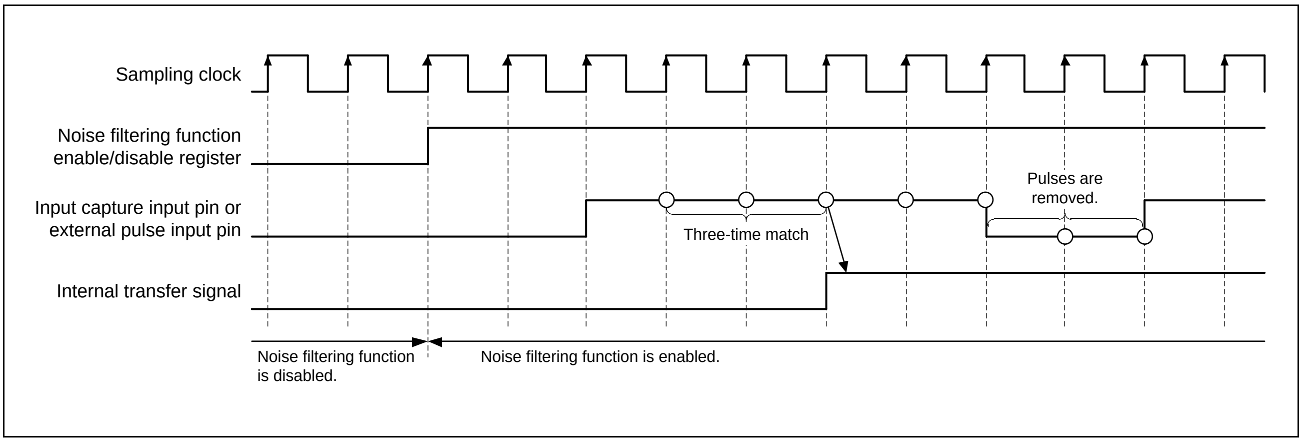


Figure 16.110 Timing of Noise Filtering

16.4 Interrupt Sources

16.4.1 Interrupt Sources and Priorities

There are three kinds of interrupt source; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own enable/disable bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt source is generated, if the corresponding enable/disable bit in TIER is set to 1, an interrupt is requested.

Relative channel priorities can be changed by the interrupt controller; however the priority within a channel is fixed. For details, refer to **Section 8, Interrupt Controller**. **Table 16.78** lists the MTU interrupt sources.

Table 16.78 MTU Interrupt Sources (1/2)

Channel	Name	Interrupt Source	DMAC Activation
MTU0	TGIA0	MTU0.TGRA input capture/compare match	Possible
	TGIB0	MTU0.TGRB input capture/compare match	Possible
	TGIC0	MTU0.TGRC input capture/compare match	Possible
	TGID0	MTU0.TGRD input capture/compare match	Possible
	TCIV0	MTU0.TCNT overflow	Not possible
	TGIE0	MTU0.TGRE compare match	Not possible
	TGIF0	MTU0.TGRF compare match	Not possible
MTU1	TGIA1	MTU1.TGRA input capture/compare match	Possible
	TGIB1	MTU1.TGRB input capture/compare match	Possible
	TCIV1	MTU1.TCNT overflow	Not possible
	TCIU1	MTU1.TCNT underflow	Not possible
MTU2	TGIA2	MTU2.TGRA input capture/compare match	Possible
	TGIB2	MTU2.TGRB input capture/compare match	Possible
	TCIV2	MTU2.TCNT overflow	Not possible
	TCIU2	MTU2.TCNT underflow	Not possible
MTU3	TGIA3	MTU3.TGRA input capture/compare match	Possible
	TGIB3	MTU3.TGRB input capture/compare match	Possible
	TGIC3	MTU3.TGRC input capture/compare match	Possible
	TGID3	MTU3.TGRD input capture/compare match	Possible
	TCIV3	MTU3.TCNT overflow	Not possible
MTU4	TGIA4	MTU4.TGRA input capture/compare match	Possible
	TGIB4	MTU4.TGRB input capture/compare match	Possible
	TGIC4	MTU4.TGRC input capture/compare match	Possible
	TGID4	MTU4.TGRD input capture/compare match	Possible
	TCIV4	MTU4.TCNT overflow/underflow*[1]	Possible
MTU5	TGIU5	MTU5.TGRU input capture/compare match	Possible
	TGIV5	MTU5.TGRV input capture/compare match	Possible
	TGIW5	MTU5.TGRW input capture/compare match	Possible
MTU6	TGIA6	MTU6.TGRA input capture/compare match	Possible
	TGIB6	MTU6.TGRB input capture/compare match	Possible
	TGIC6	MTU6.TGRC input capture/compare match	Possible
	TGID6	MTU6.TGRD input capture/compare match	Possible
	TCIV6	MTU6.TCNT overflow	Not possible

Table 16.78 MTU Interrupt Sources (2/2)

Channel	Name	Interrupt Source	DMAC Activation
MTU7	TGIA7	MTU7.TGRA input capture/compare match	Possible
	TGIB7	MTU7.TGRB input capture/compare match	Possible
	TGIC7	MTU7.TGRC input capture/compare match	Possible
	TGID7	MTU7.TGRD input capture/compare match	Possible
	TCIV7	MTU7.TCNT overflow/underflow*[1]	Possible
MTU8	TGIA8	MTU8.TGRA input capture/compare match	Possible
	TGIB8	MTU8.TGRB input capture/compare match	Possible
	TGIC8	MTU8.TGRC input capture/compare match	Possible
	TGID8	MTU8.TGRD input capture/compare match	Possible
	TCIV8	MTU8.TCNT overflow	Possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

Note 1. The underflow interrupt source is valid only in complementary PWM mode.

(1) Input Capture/Compare Match Interrupt

If the TIER.TGIE bit is set to 1 when a TGR input capture/compare match occurs on a channel, an interrupt is requested. The MTU has 33 input capture/compare match interrupts (six for MTU0, four each for MTU3, MTU4, MTU6, MTU7, and MTU8, two each for MTU1 and MTU2, and three for MTU5).

(2) Overflow Interrupt

If the TIER.TCIEV bit is set to 1 when a TCNT overflow occurs on a channel, clearing an interrupt is requested. The MTU provides a total of eight overflow interrupts (one for each channel except MTU5).

In complementary PWM mode, an overflow interrupt is generated even when an underflow occurs in TCNT in MTU4 or MTU7.

(3) Underflow Interrupt

If the TIER.TCIEU bit is set to 1 when a TCNT underflow occurs on a channel, an interrupt is requested. The MTU has two underflow interrupts (one each for MTU1, and MTU2).

16.4.2 DMAC Activation

(1) DMAC Activation

The DMAC can be activated by the TGR input capture/compare match interrupt and the overflow interrupt in MTU4 and MTU7. For details, see **Section 14, Direct Memory Access Controller**.

The MTU provides a total of 33 interrupts (input capture/compare match and overflow interrupts) that can be used as DMAC activation sources: four each for MTU0, MTU3, MTU6, and MTU8, two each for MTU1 and MTU2, five each for MTU4 and MTU7, and three for MTU5.

16.4.3 A/D Converter Activation

The A/D converter can be activated by one of the following three methods in the MTU. **Table 16.79** shows the relationship between interrupt sources and A/D converter start request signals.

(1) A/D Converter Activation by TGRA Input Capture/Compare Match or at MTU4.TCNT (MTU7.TCNT) Trough in Complementary PWM Mode

The A/D converter can be activated by the occurrence of a TGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in MTU4.TIER (MTU7.TIER) is set to 1, the A/D converter can be activated at the trough of MTU4.TCNT (MTU7.TCNT) count (MTU4.TCNT (MTU7.TCNT) = H’0000).

A/D converter start request signal TRGAnN is issued to the A/D converter under either of the following conditions (n = 0 to 4, 6, or 7).

- When a TGRA input capture/compare match occurs on a channel while the TIER.TTG bit is set to 1
- When the MTU4.TCNT (MTU7.TCNT) count reaches the trough (MTU4.TCNT (MTU7.TCNT) = H’0000) during complementary PWM operation while the TTGE2 bit in MTU4.TIER (MTU7.TIER) is set to 1

When either condition is satisfied, if A/D converter start signal TRGAnN from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(2) A/D Converter Activation by Compare Match between MTU0.TCNT and MTU0.TGRE

A/D converter start request signal TRG0N is issued to the A/D converter when a compare match occurs between MTU0.TCNT and MTU0.TGRE.

When a compare match occurs between MTU0.TCNT and MTU0.TGRE in MTU0 while the TTGE2 bit.TIER2 is set to 1, A/D converter start request TGR0N is issued to the A/D converter. If A/D converter start signal TRG0N from the MTU is selected as the trigger in the A/D converter, A/D conversion will start.

(3) A/D Converter Activation by A/D Converter Start Request Delaying Function

The A/D converter can be activated by generating A/D converter start request signal TRG4AN or TRG4BN (TRG7AN or TRG7BN) when the MTU4.TCNT (MTU7.TCNT) count matches the MTU4.TADCORA or MTU4.TADCORB (MTU7.TADCORA or MTU7.TADCORB) value if the UT4AE, DT4AE, UT4BE, or DT4BE (UT7AE, DT7AE, UT7BE, or DT7BE) bit in the A/D converter start request control register (MTU4.TADCR (MTU7.TADCR)) is set to 1. For details, refer to **Section 16.3.9, A/D Converter Start Request Delaying Function**.

A/D conversion will start when TRG4AN (TRG7AN) is generated if A/D converter start signal TRG4AN (TRG7AN) from the MTU is selected as the trigger in the A/D converter, when TRG4BN (TRG7BN) is generated if TRG4BN (TRG7BN) from the MTU is selected as the trigger in the A/D converter, or when TRG4ABN (TRG7ABN) is generated if TRG4ABN (TRG7ABN) from the MTU is selected as the trigger in the A/D converter.

Table 16.79 Interrupt Sources and A/D Converter Start Request Signals

Target Registers	Interrupt Source	A/D Converter Start Request Signal
MTU0.TGRA and MTU0.TCNT	Input capture/compare match	TRGA0N
MTU1.TGRA and MTU1.TCNT		TRGA1N
MTU2.TGRA and MTU2.TCNT		TRGA2N
MTU3.TGRA and MTU3.TCNT		TRGA3N
MTU4.TGRA and MTU4.TCNT*1		TRGA4N
MTU4.TCNT	MTU4.TCNT trough in complementary PWM mode	
MTU6.TGRA and MTU6.TCNT	Input capture/compare match	TRGA6N
MTU7.TGRA and MTU7.TCNT*1		TRGA7N
MTU7.TCNT	MTU7.TCNT trough in complementary PWM mode	
MTU0.TGRE and MTU0.TCNT	Compare match	TRG0N
MTU4.TADCORA and MTU4.TCNT		TRG4AN
MTU4.TADCORB and MTU4.TCNT		TRG4BN
MTU7.TADCORA and MTU7.TCNT		TRG7AN
MTU7.TADCORB and MTU7.TCNT		TRG7BN
MTU4.TADCORA and MTU4.TCNT, MTU4.TADCORB and MTU4.TCNT	Compare match (interrupt skipping function 2)	TRG4ABN
MTU7.TADCORA and MTU7.TCNT, MTU7.TADCORB and MTU7.TCNT		TRG7ABN

Note 1. Since PWM waveforms are generated in complementary PWM mode, MTU4.TGRA (MTU7.TGRA) compare match not only with MTU4.TCNT (MTU7.TCNT) but also with MTU3.TCNT (MTU6.TCNT) and TCNTSA (TCNTSB) is detected. Accordingly, when compare match with MTU3.TCNT (MTU6.TCNT) and TCNTSA (TCNTSB) occurs, TRGA4N (TRGA7N) is also generated.

When MTU3 and MTU 4 (MTU 6 and MTU7) are made to operate in complementary PWM mode for generating an A/D converter start request, use the A/D converter start request by compare match between MTU4.TCNT (MTU7.TCNT) and MTU4.TADCORA or TADCORB (MTU7.TADCORA or TADCORB).

16.5 Operation Timing

16.5.1 Input/Output Timing

(1) TCNT Count Timing

Figure 16.111 and **Figure 16.112** show the TCNT count timing in internal clock operation, **Figure 16.113** shows the TCNT count timing in external clock operation (normal mode), and **Figure 16.114** shows the TCNT count timing in external clock operation (phase counting mode).

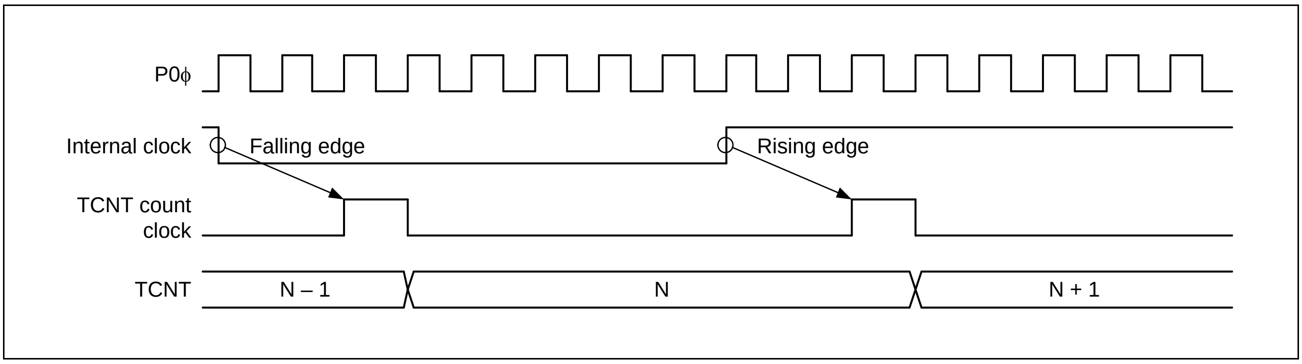


Figure 16.111 Count Timing in Internal Clock Operation (MTU0 to MTU4 and MTU6 to MTU8)

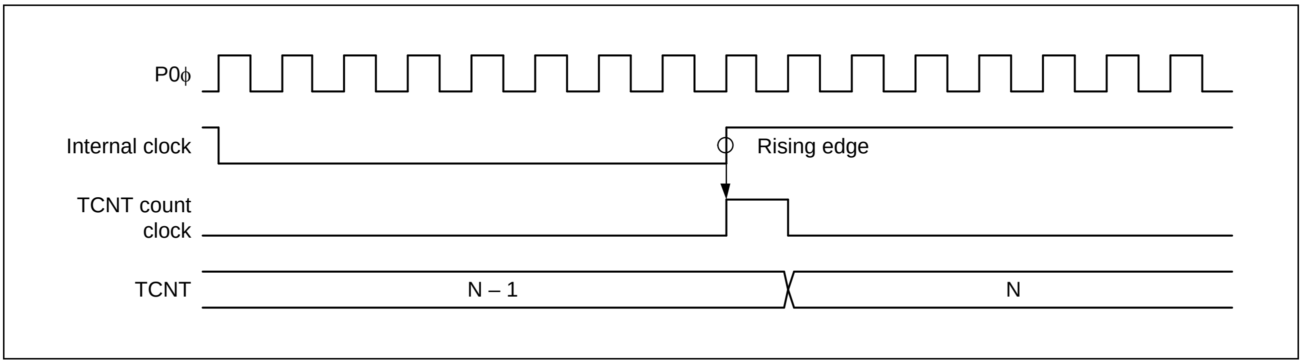


Figure 16.112 Count Timing in Internal Clock Operation (MTU5)

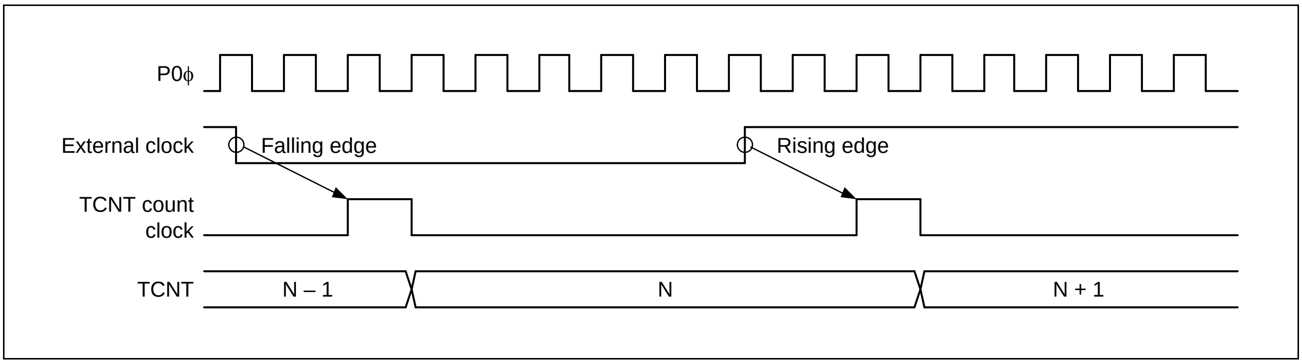


Figure 16.113 Count Timing in External Clock Operation (MTU0 to MTU4, MTU6 to MTU8)

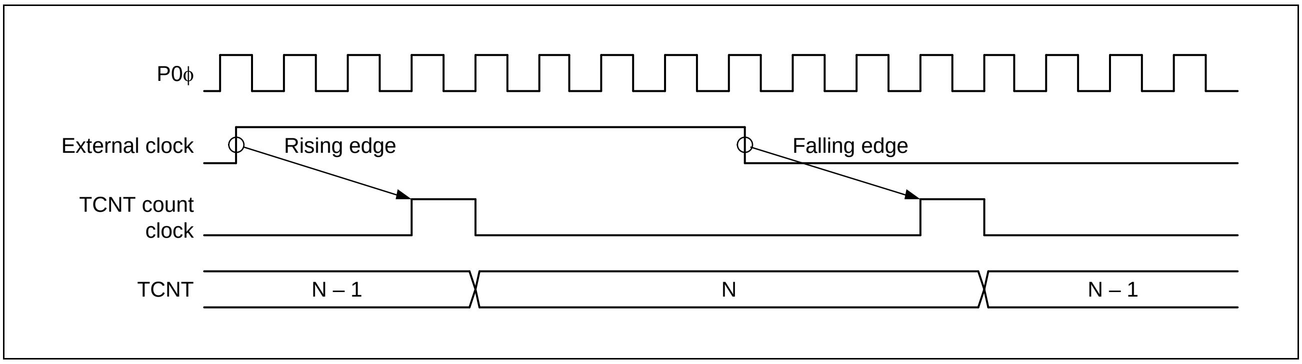


Figure 16.114 Count Timing in External Clock Operation (Phase Counting Mode)

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the value set in TIOR is output from MTIOCnm pin (n = 0 to 4, 6, 7, 8; m = A to D). After a match between TCNT and TGR, the compare match signal is not generated until the TCNT count clock is generated.

Figure 16.115 shows the output compare output timing (normal mode or PWM mode) and **Figure 16.116** shows the output compare output timing (complementary PWM mode or reset-synchronized PWM mode).

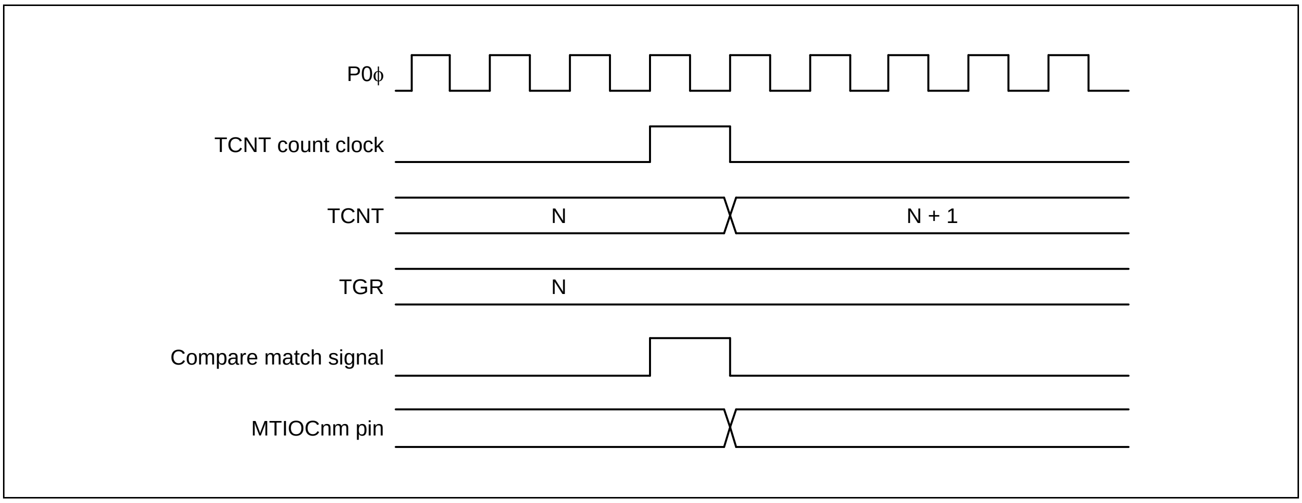


Figure 16.115 Output Compare Output Timing (Normal Mode or PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D)

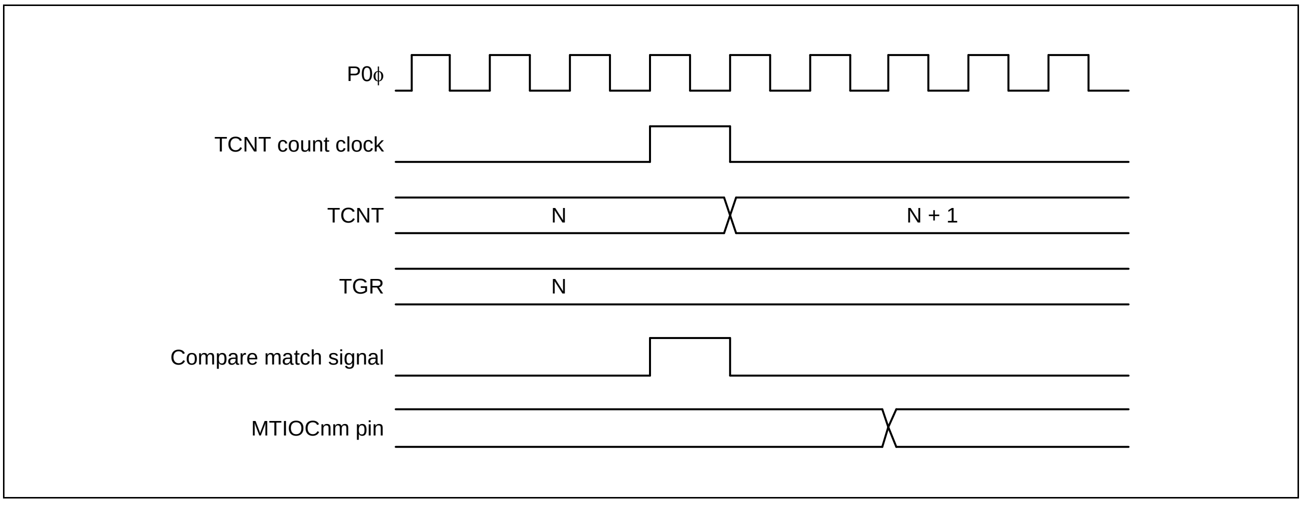


Figure 16.116 Output Compare Output Timing (Complementary PWM Mode or Reset-Synchronized PWM Mode) (n = 0 to 4, 6, 7, 8; m = A to D)

(3) Input Capture Signal Timing

Figure 16.117 shows the input capture signal timing.

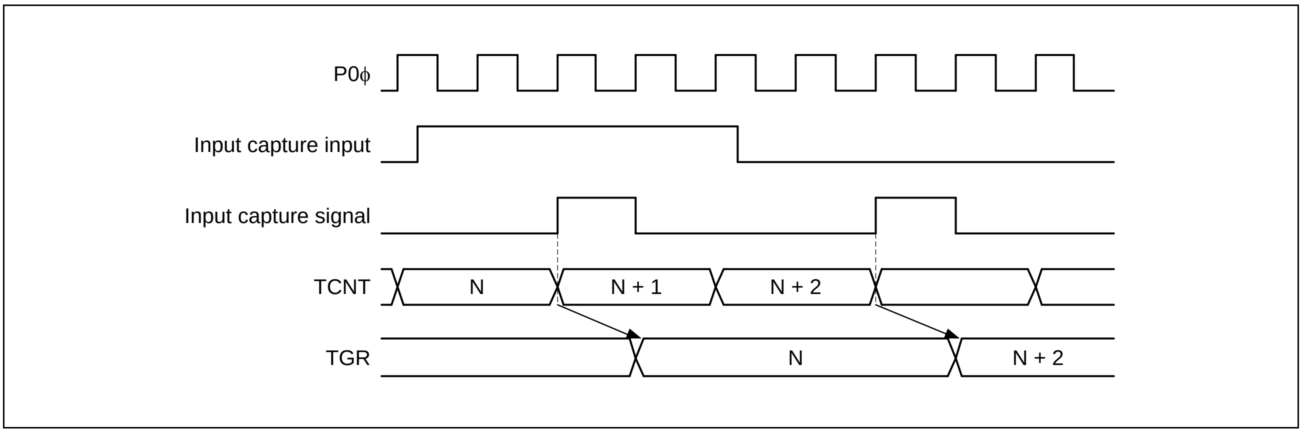


Figure 16.117 Input Capture Input Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 16.118 and **Figure 16.119** show the timing when counter clearing on compare match is specified, and **Figure 16.120** shows the timing when counter clearing on input capture is specified.

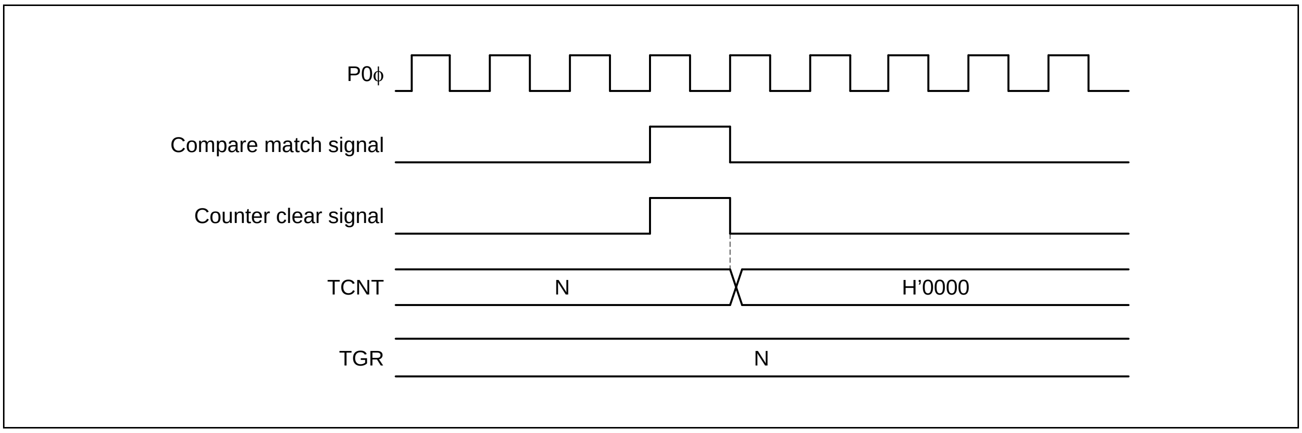


Figure 16.118 Counter Clear Timing (Compare Match) (MTU0 to MTU4 and MTU6 to MTU8)

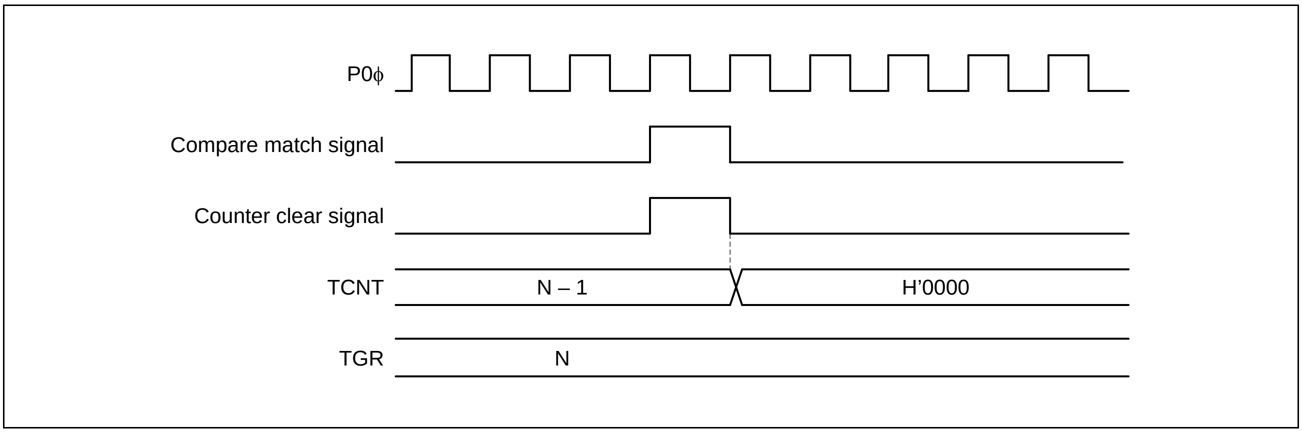


Figure 16.119 Counter Clear Timing (Compare Match) (MTU5)

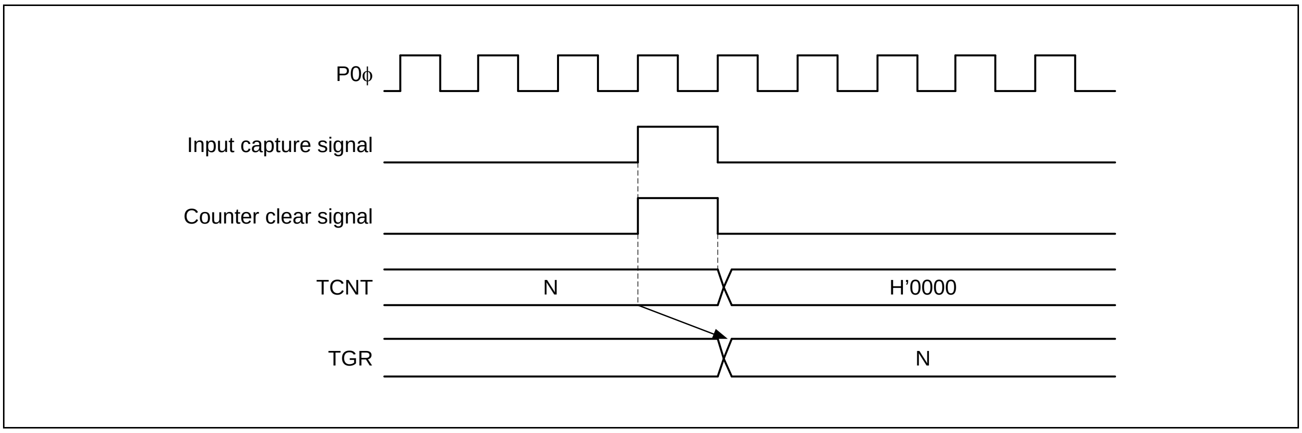


Figure 16.120 Counter Clear Timing (Input Capture) (MTU0 to MTU8)

(5) Buffer Operation Timing

Figure 16.121 to **Figure 16.123** show the timing in buffer operation.

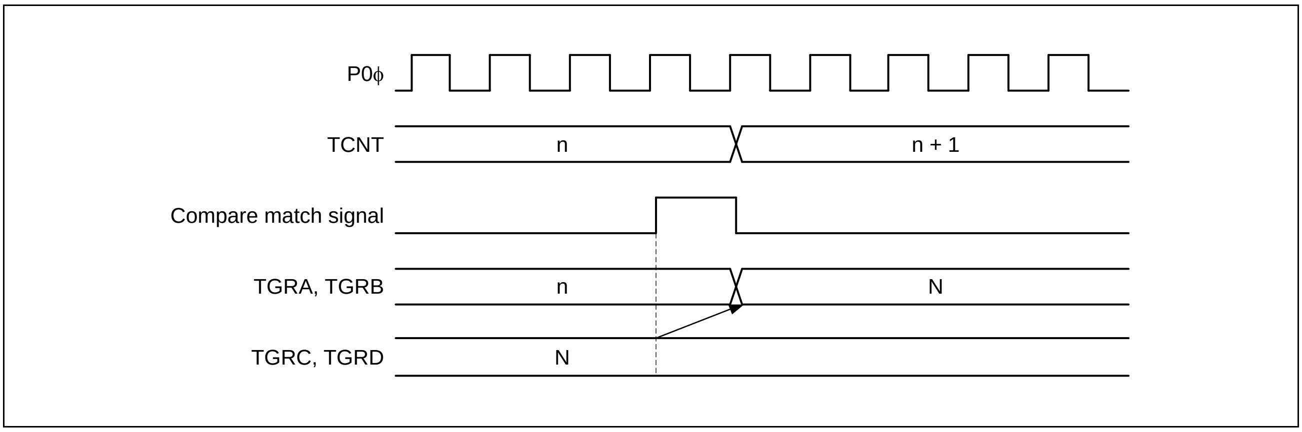


Figure 16.121 Buffer Operation Timing (Compare Match)

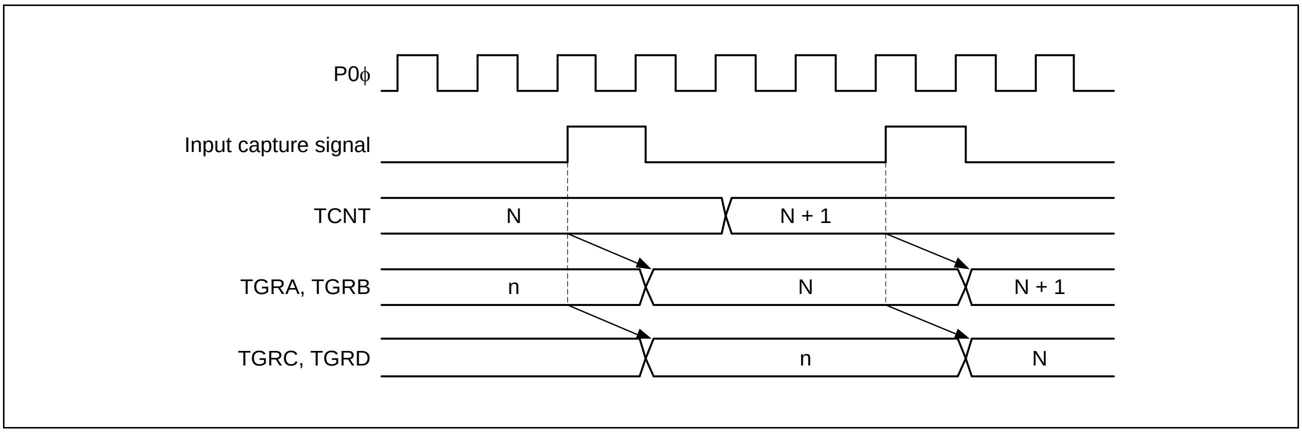


Figure 16.122 Buffer Operation Timing (Input Capture)

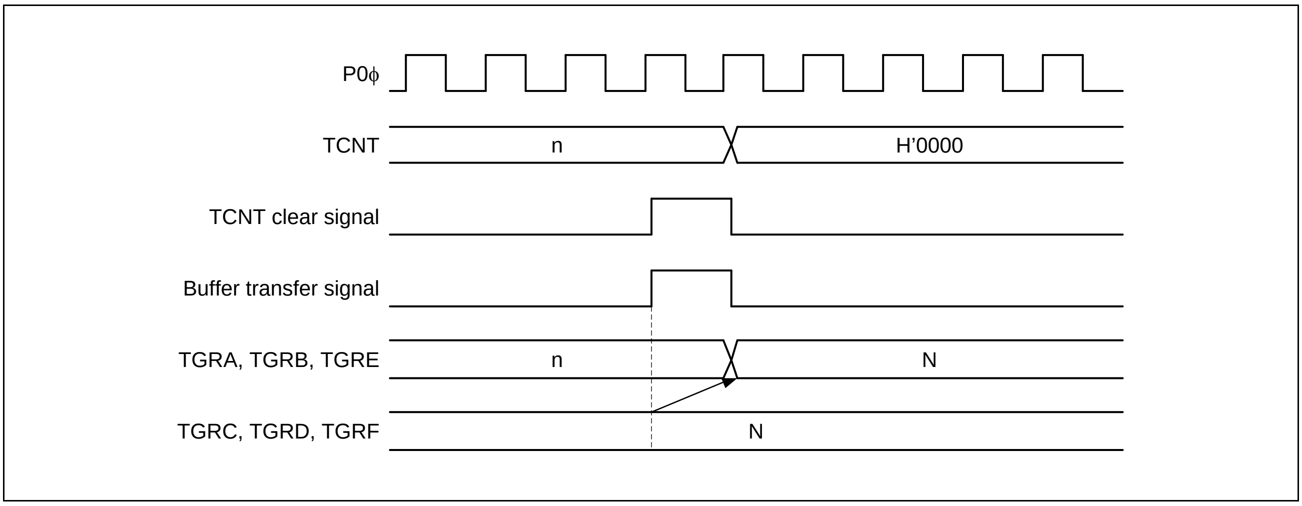


Figure 16.123 Buffer Operation Timing (When TCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figure 16.124 to **Figure 16.126** show the buffer transfer timing in complementary PWM mode.

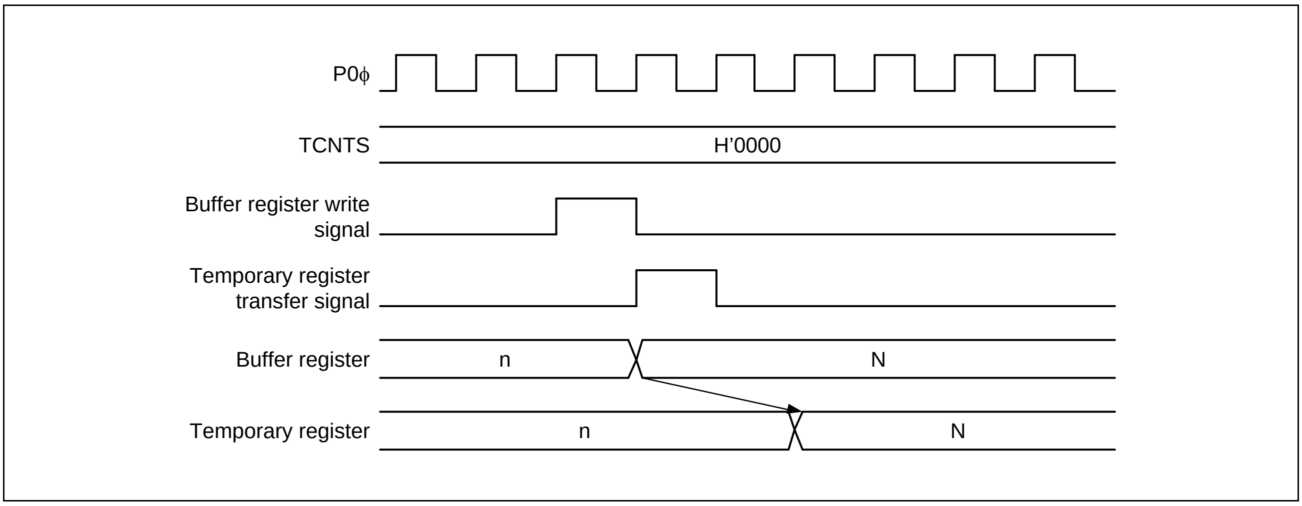


Figure 16.124 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Stopped)

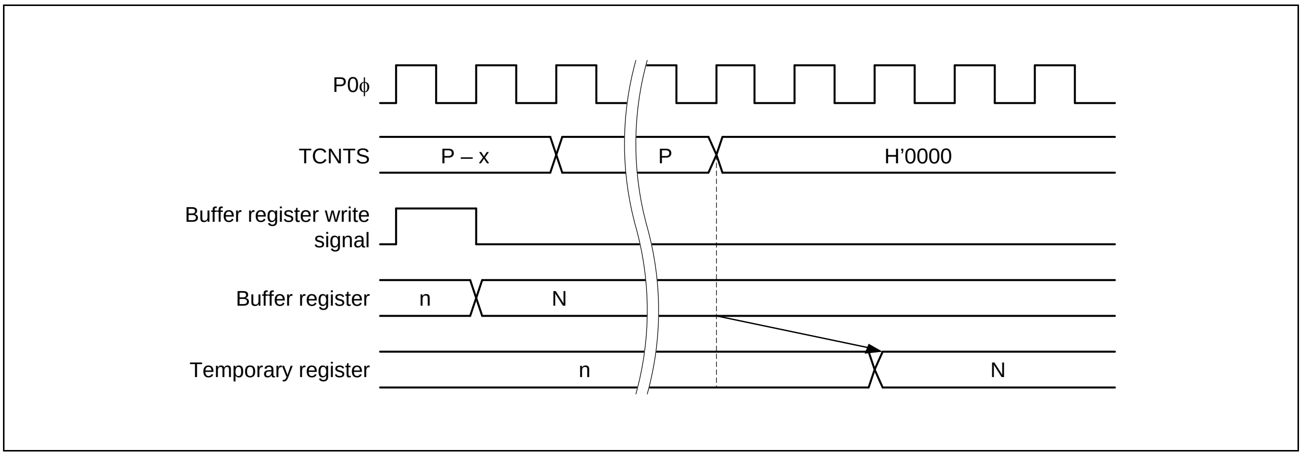


Figure 16.125 Transfer Timing from Buffer Register to Temporary Register (TCNTSA Operating)

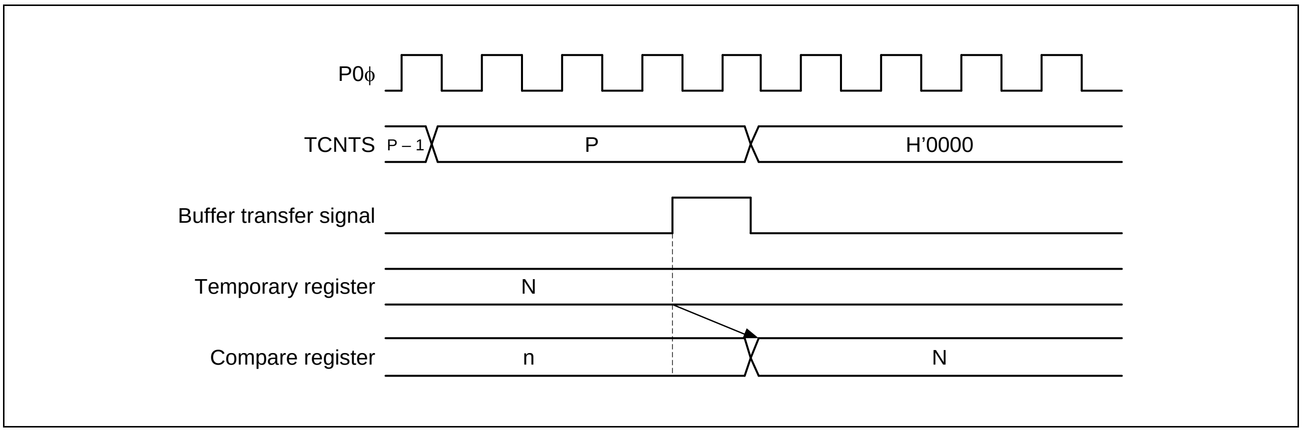


Figure 16.126 Transfer Timing from Temporary Register to Compare Register

16.5.2 Interrupt Signal Timing

(1) TGI Interrupt Timing by Compare Match

Figure 16.127 and **Figure 16.128** show the TGI interrupt request signal timing when a compare match occurs.

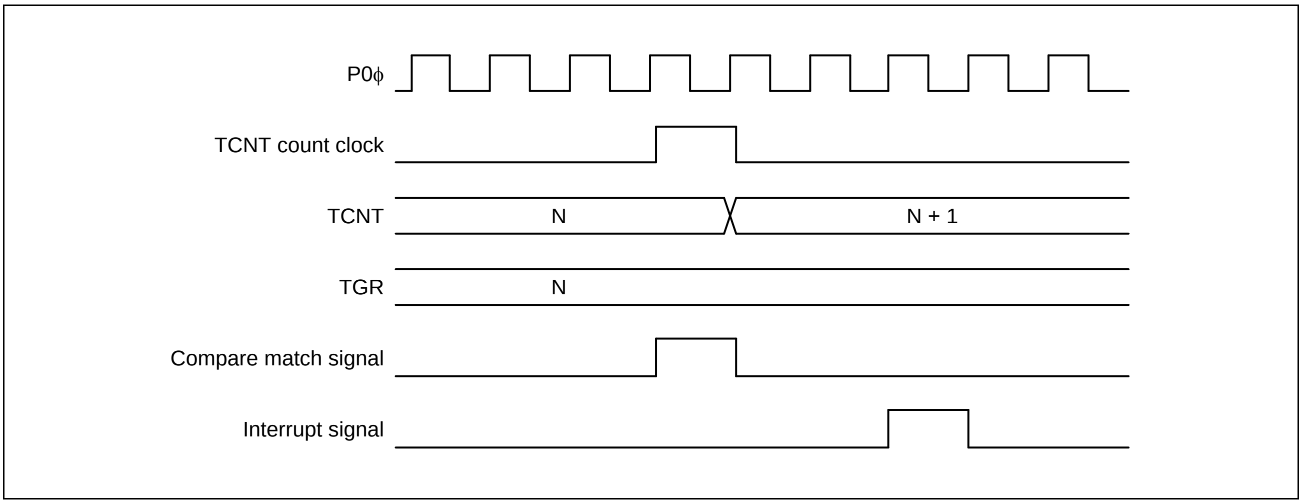


Figure 16.127 TGI Interrupt Timing (Compare Match) (MTU0 to MTU4 and MTU6 to MTU8)

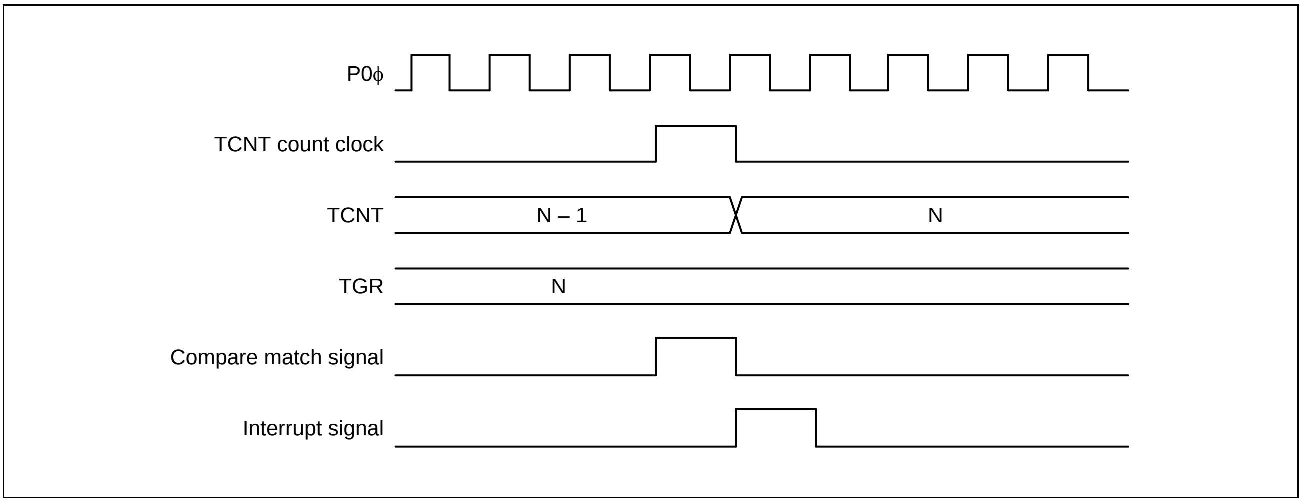


Figure 16.128 TGI Interrupt Timing (Compare Match) (MTU5)

(2) TGI Interrupt Timing by Input Capture

Figure 16.129 and **Figure 16.130** show the TGI interrupt request signal timing when an input capture occurs.

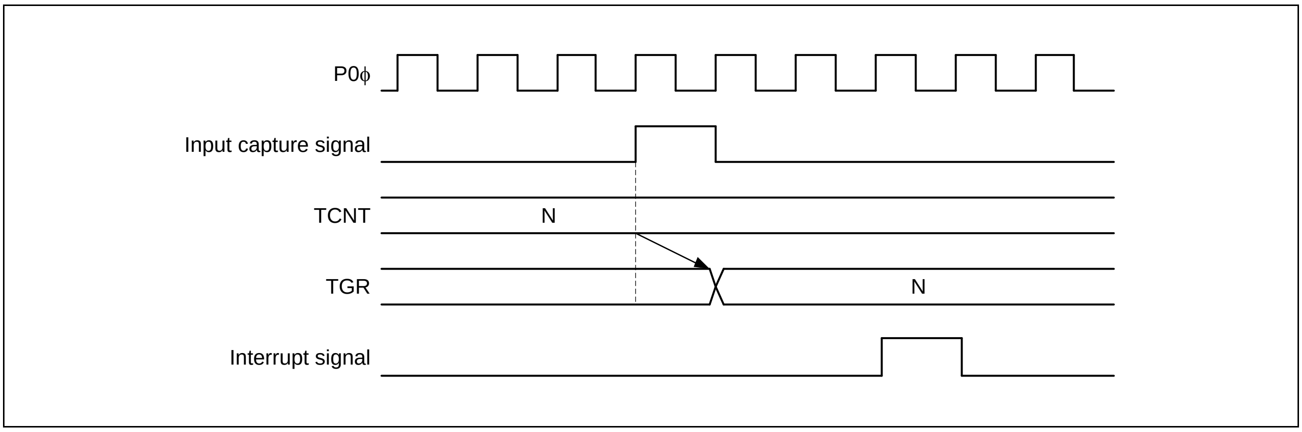


Figure 16.129 TGI Interrupt Timing (Input Capture) (MTU0 to MTU4 and MTU6 to MTU8)

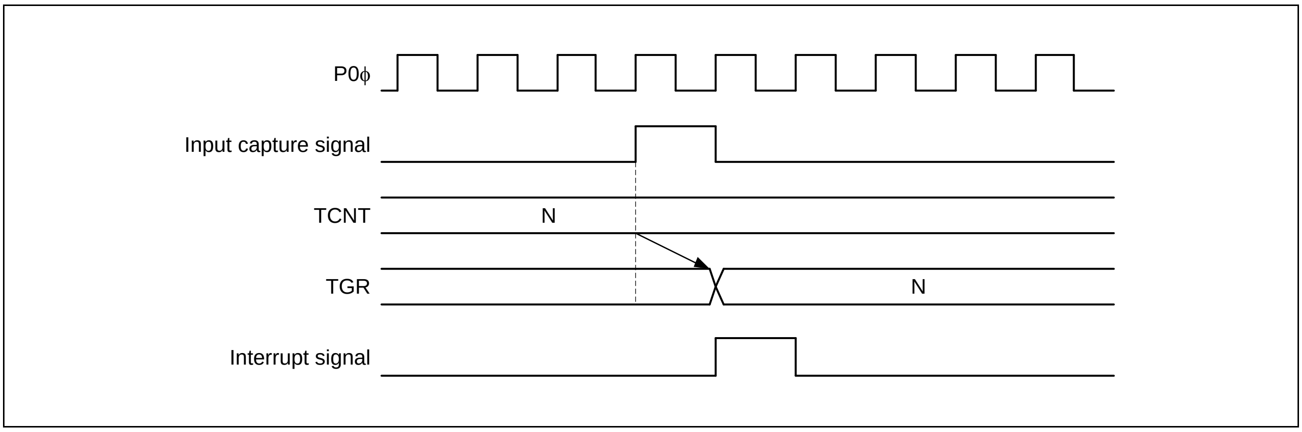


Figure 16.130 TGI Interrupt Timing (Input Capture) (MTU5)

(3) TCFV and TCFU Interrupt Timing

Figure 16.131 shows the TCIV interrupt request signal timing when an overflow is generated.

Figure 16.132 shows the TCIU interrupt request signal timing when an underflow is generated.

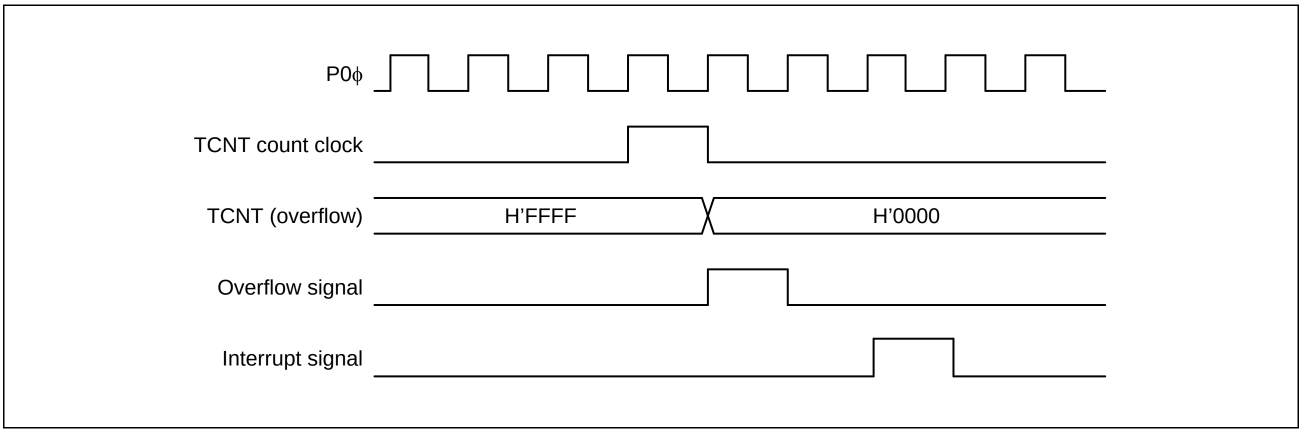


Figure 16.131 TCIV Interrupt Timing

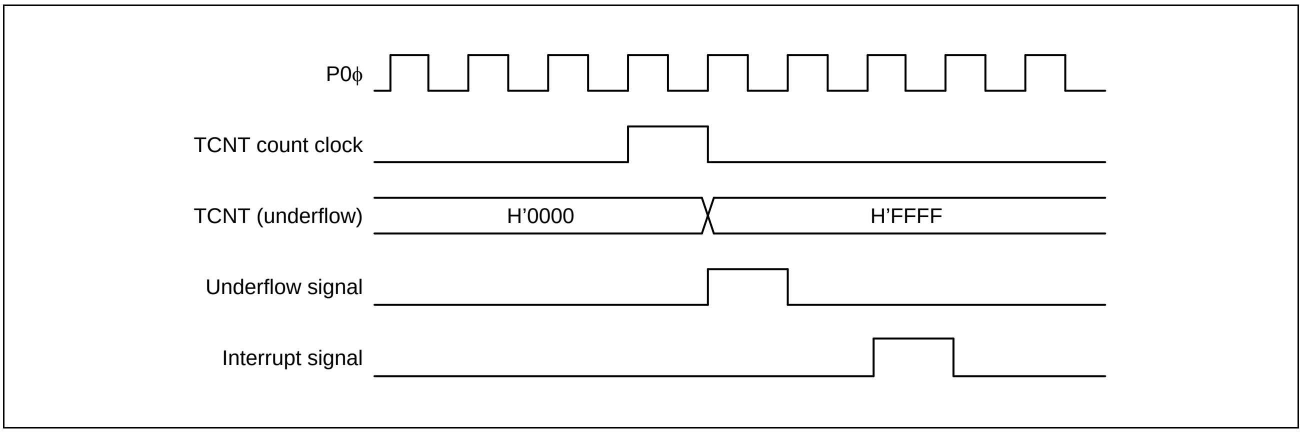


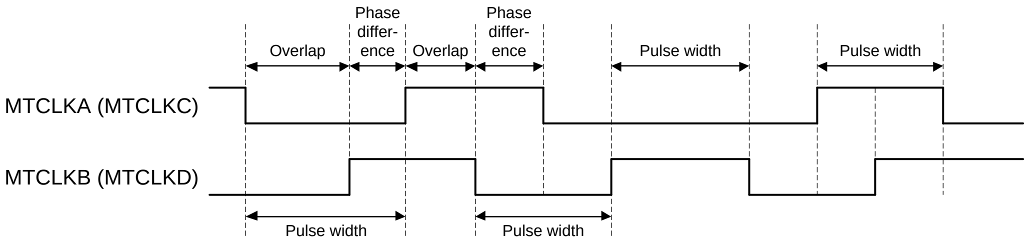
Figure 16.132 TCIU Interrupt Timing

16.6 Usage Note

16.6.1 Count Clock Restrictions

The count clock source pulse width must be at least three P0ϕ clock cycles for single-edge detection, and at least five P0ϕ clock cycles for both-edge detection. The MTU will not operate properly at narrower pulse widths.

In phase counting mode, the phase difference and overlap between the two input clocks must be at least three P0ϕ clock cycles, and the pulse width must be at least 5P0ϕ clock cycles. **Figure 16.133** shows the input clock conditions in phase counting mode.



Note: Phase difference and overlap: 3 P0ϕ clock cycles or more

Note: Pulse width: 5 P0ϕ clock cycles or more

Figure 16.133 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

16.6.2 Note on Cycle Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which TCNU updates the matched count value). Consequently, the actual counter frequency is given by the following formula:

- MTU0 to MTU4 and MTU6 to MTU8

$$f = \frac{\mathrm{CNTCLK}}{\mathrm{N}+1}$$

- MTU5

$$f = \frac{\mathrm{CNTCLK}}{\mathrm{N}}$$

f: Counter frequency

CNTCLK: The count clock frequency set by TCR.TPSC[2:0] and TCR2.TPSC2[2:0]

N: TGR setting

16.6.3 Contention between TCNT Write and Clear Operations

If the counter clear signal is generated in the TCNT write cycle, TCNT clearing takes precedence and TCNT write operation is not performed.

Figure 16.134 shows the timing in this case.

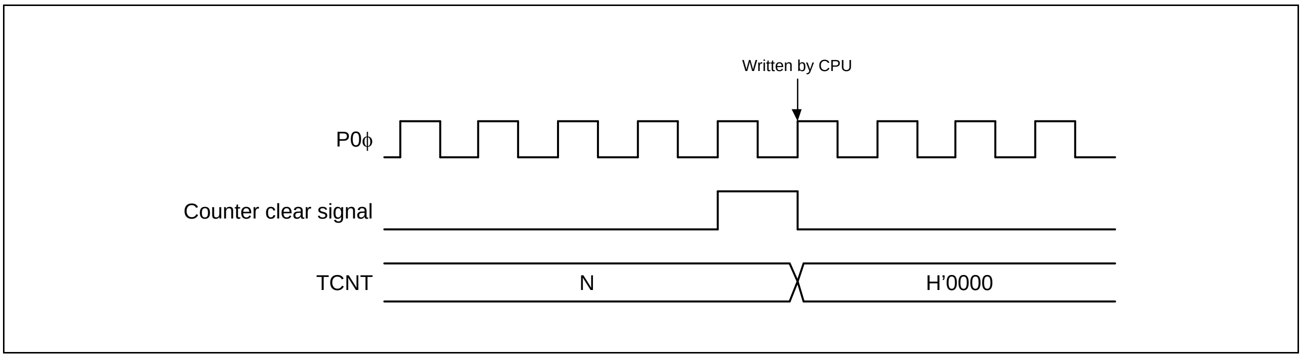


Figure 16.134 Contention between TCNT Write and Clear Operations

16.6.4 Contention between TCNT Write and Increment Operations

If incrementing occurs in a TCNT write cycle, TCNT write operation takes precedence and TCNT is not incremented.

Figure 16.135 shows the timing in this case.

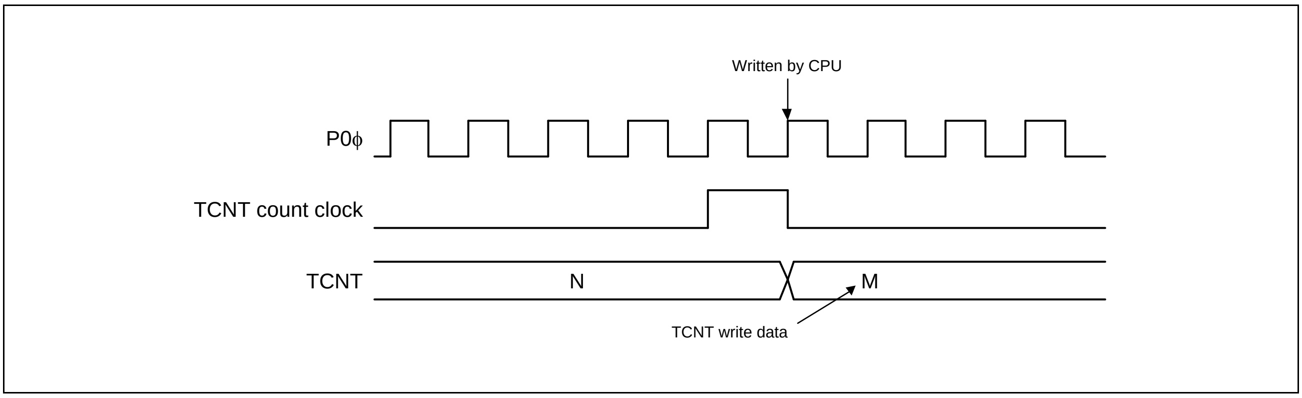


Figure 16.135 Contention between TCNT Write and Increment Operations

16.6.5 Contention between TGR Write Operation and Compare Match

If a compare match occurs in a TGR write cycle, TGR write operation is executed and the compare match signal is also generated.

Figure 16.136 shows the timing in this case.

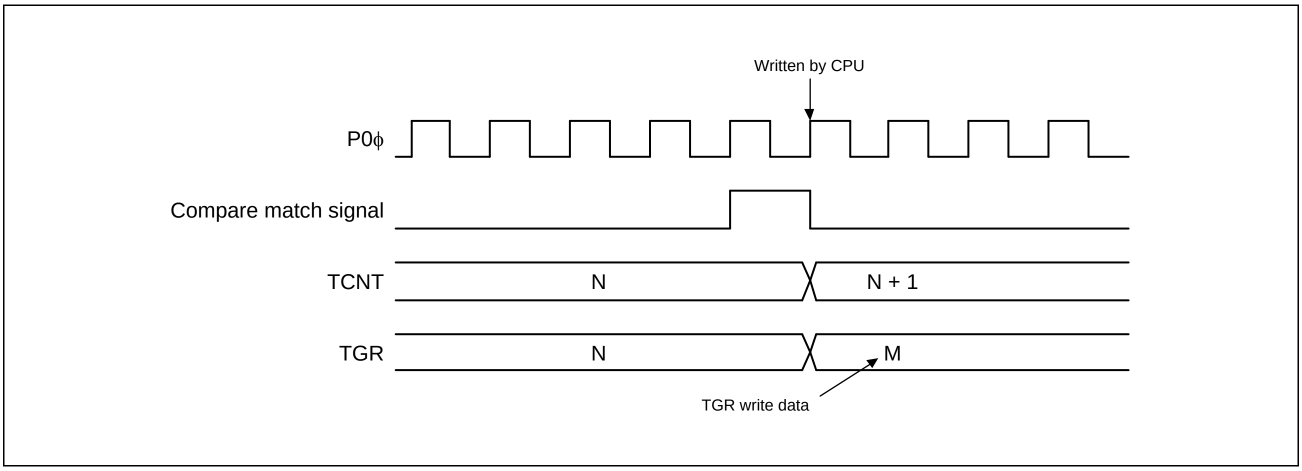


Figure 16.136 Contention between TGR Write Operation and Compare Match

16.6.6 Contention between Buffer Register Write Operation and Compare Match

If a compare match occurs in the T2 state in a TGR write cycle, the data before write operation is transferred to TGR by the buffer operation.

Figure 16.137 shows the timing in this case.

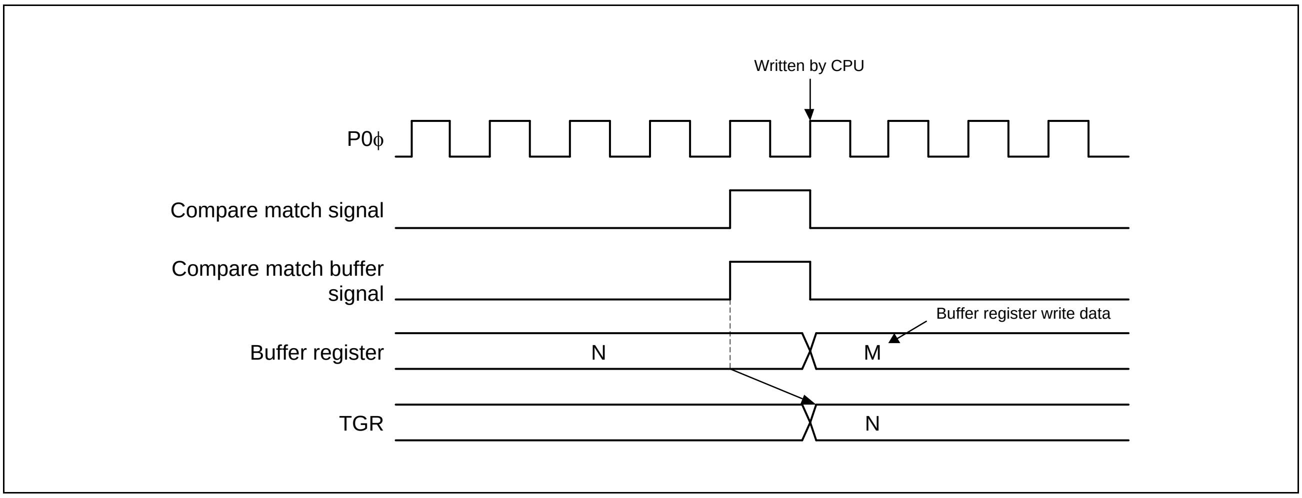


Figure 16.137 Contention between Buffer Register Write Operation and Compare Match

16.6.7 Contention between Buffer Register Write and TCNT Clear Operations

When the buffer transfer timing is set to the TCNT clear timing by the timer buffer transfer mode register (TBTM), if TCNT is cleared during the TGR write cycle, the data before write operation is transferred to TGR by the buffer operation.

Figure 16.138 shows the timing in this case.

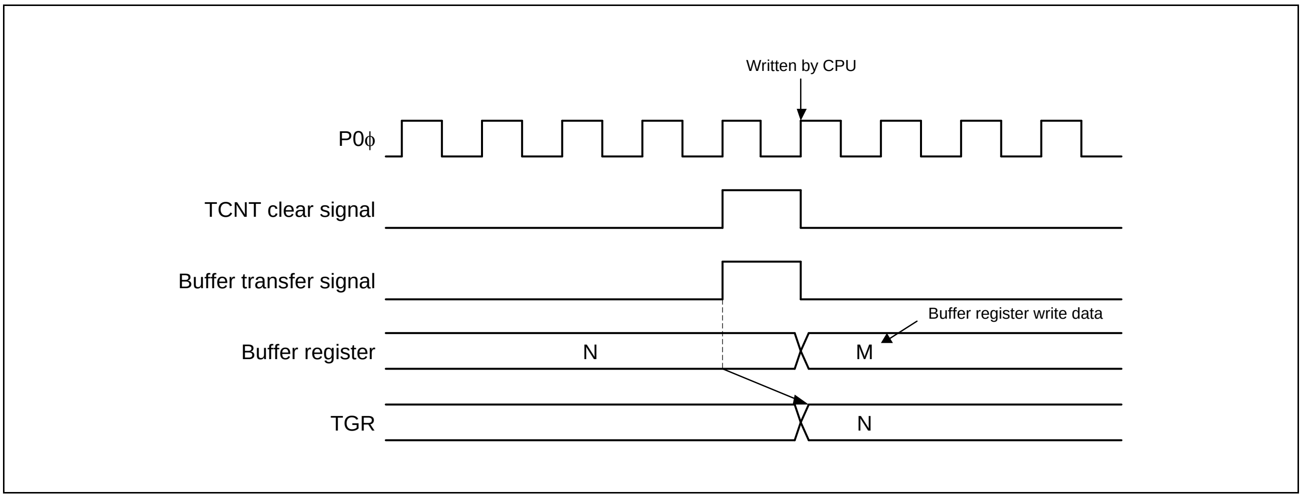


Figure 16.138 Contention between Buffer Register Write and TCNT Clear Operations

16.6.8 Contention between TGR Read Operation and Input Capture

If an input capture signal is generated in a TGR read cycle, the data before input capture transfer is read.

Figure 16.139 shows the timing in this case.

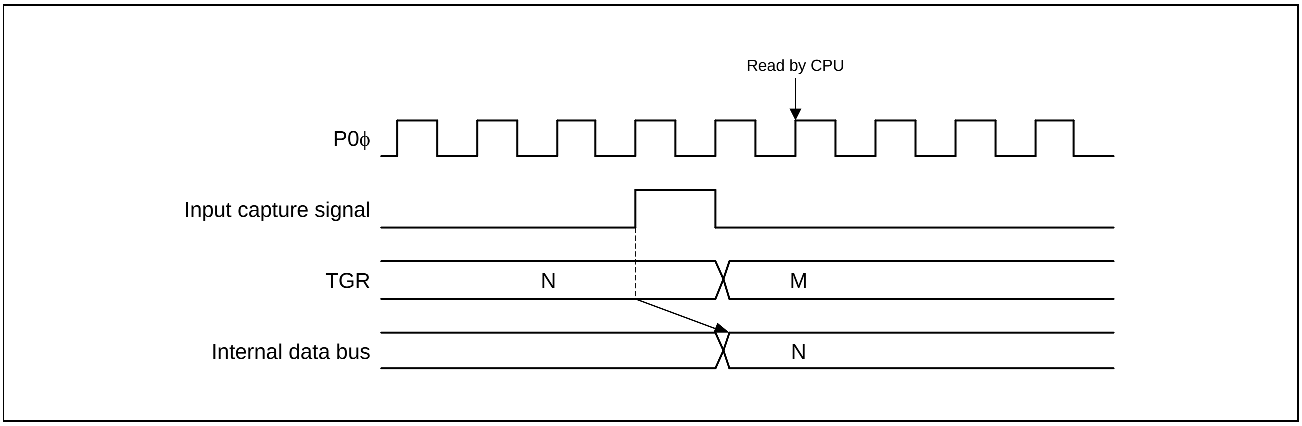


Figure 16.139 Contention between TGR Read Operation and Input Capture (MTU0 to MTU8)

16.6.9 Contention between TGR Write Operation and Input Capture

If an input capture signal is generated in the TGR write cycle, the input capture operation takes precedence and the TGR write operation is not performed in MTU0 to MTU4 and MTU6 to MTU8. In MTU5, the TGR write operation is performed and the input capture signal is generated.

Figure 16.140 and **Figure 16.141** show the timing in this case.

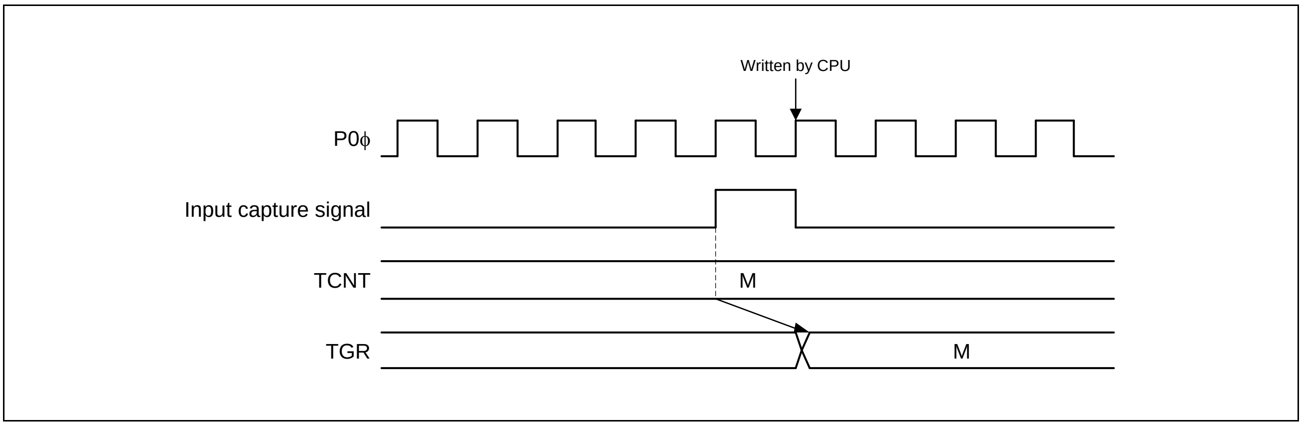


Figure 16.140 Contention between TGR Write Operation and Input Capture (MTU0 to MTU4 and MTU6 to MTU8)

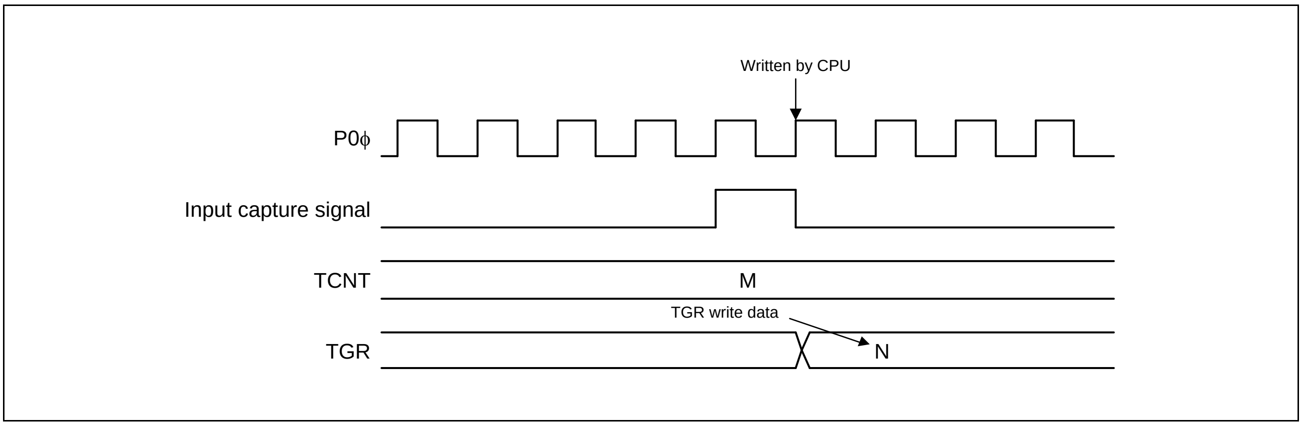


Figure 16.141 Contention between TGR Write Operation and Input Capture (MTU5)

16.6.10 Contention between Buffer Register Write Operation and Input Capture

If an input capture signal is generated in the buffer register write cycle, the buffer operation takes precedence and the buffer register write operation is not performed.

Figure 16.142 shows the timing in this case.

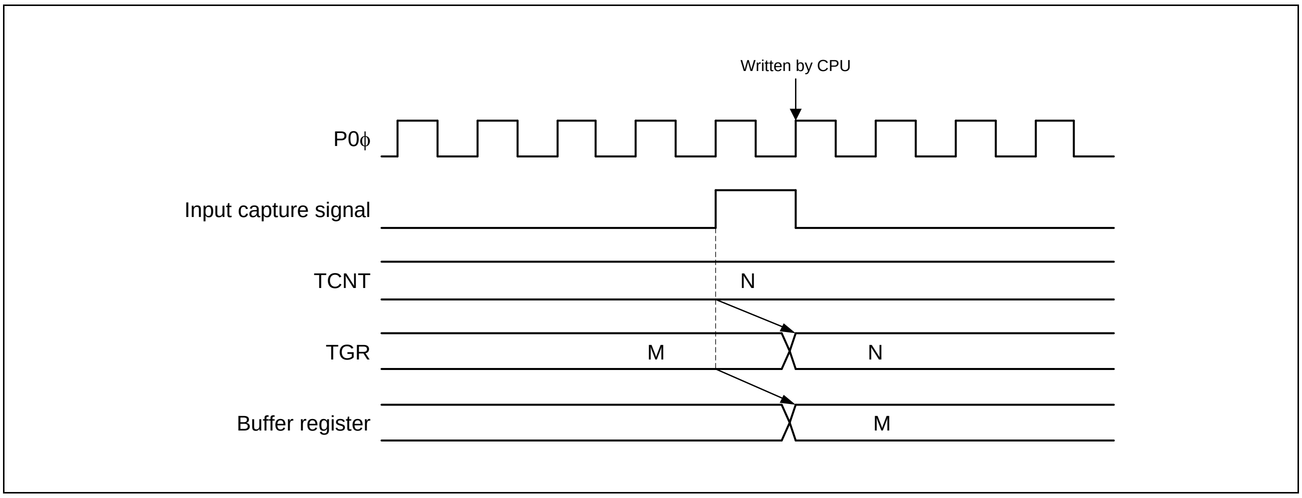


Figure 16.142 Contention between Buffer Register Write Operation and Input Capture

16.6.11 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

With timer counters TCNT in MTU1 and MTU2 cascaded, when a contention occurs between the counting by TCNT in MTU1 (a TCNT overflow/underflow in MTU2) and the writing to TCNT in MTU2, the TCNT write operation is performed in MTU2 and the TCNT count signal in MTU1 is disabled. In this case, if TGRA in MTU1 works as a compare match register and there is a match between the values of TGRA and TCNT in MTU1, a compare match signal is issued.

Furthermore, when the TCNT count clock in MTU1 is selected as the input capture source of MTU0, TGRA to TGRD in MTU0 work in input capture mode. In addition, when the TGRC compare match/input capture in MTU0 is selected as the input capture source of TGRB in MTU1, TGRB in MTU1 works in input capture mode.

Figure 16.143 shows the timing in this case.

When setting the TCNT clearing function in cascaded operation, be sure to synchronize MTU1 and MTU2.

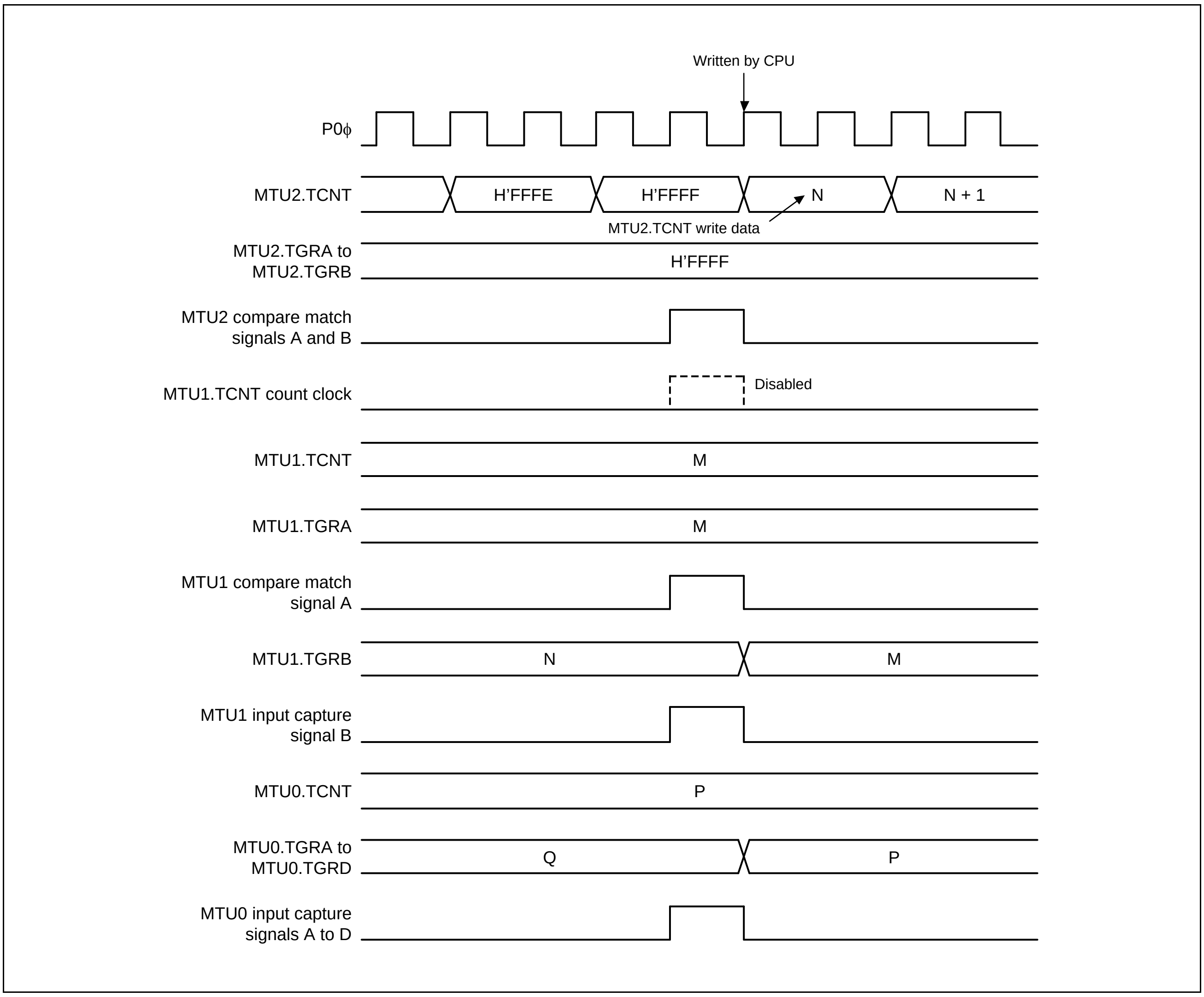


Figure 16.143 Contention between MTU2.TCNT Write Operation and Overflow/Underflow in Cascaded Operation

16.6.12 Counter Value When Count Operation is Stopped in Complementary PWM Mode

When counting operation in MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT) is stopped in complementary PWM mode, the MTU3.TCNT (MTU6.TCNT) value is set to the timer dead time register (TDDRA (TDDRB)) value and MTU4.TCNT (MTU7.TCNT) is set to H’0000.

When operation is restarted in complementary PWM mode, counting begins automatically from the initial setting state.

Figure 16.144 shows this operation.

When counting begins in another operating mode, be sure to make initial settings in MTU3.TCNT and MTU4.TCNT (MTU6.TCNT and MTU7.TCNT).

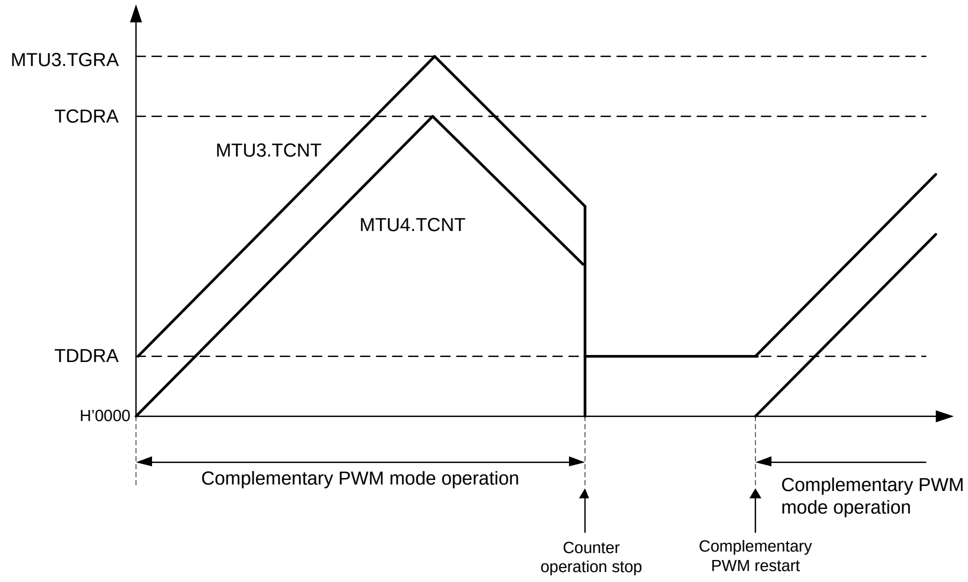


Figure 16.144 Counter Value When Stopped in Complementary PWM Mode

16.6.13 Buffer Operation Setting in Complementary PWM Mode

When modifying the PWM cycle set register (MTU3.TGRA or MTU6.TGRA), timer cycle data register (TCDRA or TCDRB), and duty set registers (MTU3.TGRB, MTU4.TGRA, and MTU4.TGRB (MTU6.TGRB, MTU7.TGRA, and MTU7.TGRB)) in complementary PWM mode, be sure to use buffer operation. In addition, set the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) to 0. The MTIOC4C (MTIOC7C) pin cannot output waveforms if the BFA bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1. Likewise, the MTIOC4D (MTIOC7D) pin cannot output waveforms if the BFB bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1.

In complementary PWM mode, buffer operation in MTU3 and MTU4 (or MTU6 and MTU7) depends on the settings in bits BFA and BFB of MTU3.TMDR1 (MTU6.TMDR1). When the BFA bit in MTU3.TMDR1 (MTU6.TMDR1) is set to 1, MTU3.TGRC (MTU6.TGRC) functions as a buffer register for MTU3.TGRA (MTU6.TGRA). At the same time, MTU4.TGRC (MTU7.TGRC) functions as a buffer register for MTU4.TGRA (MTU7.TGRA), and TCBRA (TCBRB) functions as a buffer register for TCDRA (TCDRB).

16.6.14 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

When setting buffer operation in reset-synchronized PWM mode, set the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) to 0. The MTIOC4C (MTIOC7C) pin cannot output waveforms if the BFA bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1. Likewise, the MTIOC4D (MTIOC7D) pin cannot output waveforms if the BFB bit in MTU4.TMDR1 (MTU7.TMDR1) is set to 1.

In reset-synchronized PWM mode, buffer operation in MTU3 and MTU4 (or MTU 6 and MTU7) depends on the settings in the BFA and BFB bits of MTU3.TMDR1 (MTU6.TMDR1). For example, if the BFA bit in MTU3.TMDR1 (MTU6.TMDR1) is set to 1, MTU3.TGRC (MTU6.TGRC) functions as a buffer register for MTU3.TGRA (MTU6.TGRA). At the same time, MTU4.TGRC (MTU7.TGRC) functions as a buffer register for MTU4.TGRA (MTU7.TGRA).

While the MTU3.TGRC (MTU6.TGRC) and MTU3.TGRD (MTU6.TGRD) are operating as buffer registers, a TGImn interrupt (m = C or D; n = 3, 4, 6, or 7) is not generated.

Figure 16.145 shows an example of MTU3.TGR (MTU6.TGR), MTU4.TGR (MTU7.TGR), MTIOC3 (MTIOC6), and MTIOC4 (MTIOC7) operation with the BFA and BFB bits in MTU3.TMDR1 (MTU6.TMDR1) set to 1 and the BFA and BFB bits in MTU4.TMDR1 (MTU7.TMDR1) set to 0.

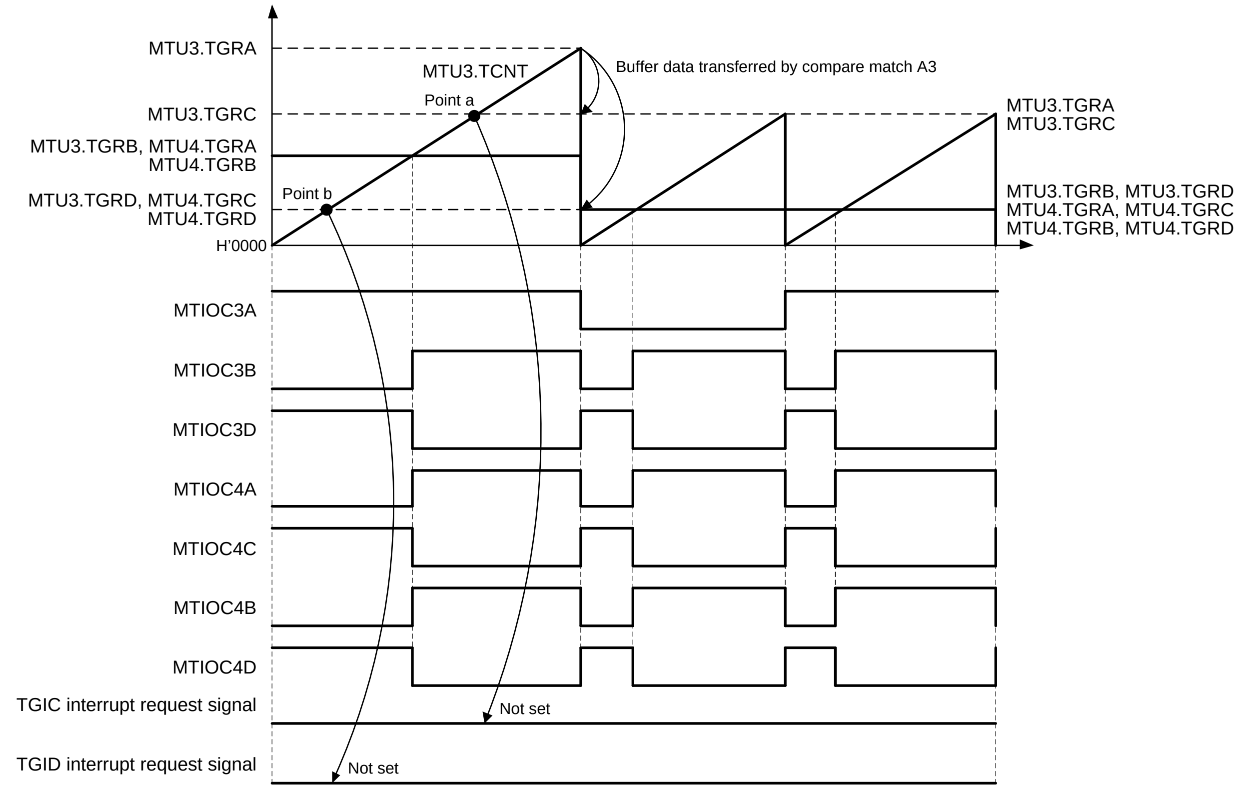


Figure 16.145 Buffer Operation and Compare Match in Reset-Synchronized PWM Mode

16.6.15 Overflow in Reset-Synchronized PWM Mode

After reset-synchronized PWM mode is selected, the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) start counting when the CST3 (CST6) bit of TSTRA (TSTRB) is set to 1. In this state, the TCNT count clock source and count edge in MTU4 (MTU7) are determined by the setting of TCR in MTU3 (MTU6).

In reset-synchronized PWM mode, with cycle register TGRA in MTU3 (MTU6) set to H’FFFF and the TGRA compare match in MTU3 (MTU6) selected as the counter clearing source, the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) count up to H’FFFF, then a compare match occurs with TGRA in MTU3 (MTU6), and the TCNT counters in MTU3 and MTU4 (MTU6 and MTU7) are both cleared. In this case, a TCIVn interrupt (n = 3, 4, 6, or 7) is not generated.

Figure 16.146 shows an example of the operation in reset-synchronized PWM mode with cycle register TGRA in MTU3 (MTU6) set to H’FFFF and the TGRA compare match in MTU3 (MTU6) specified for the counter clearing source.

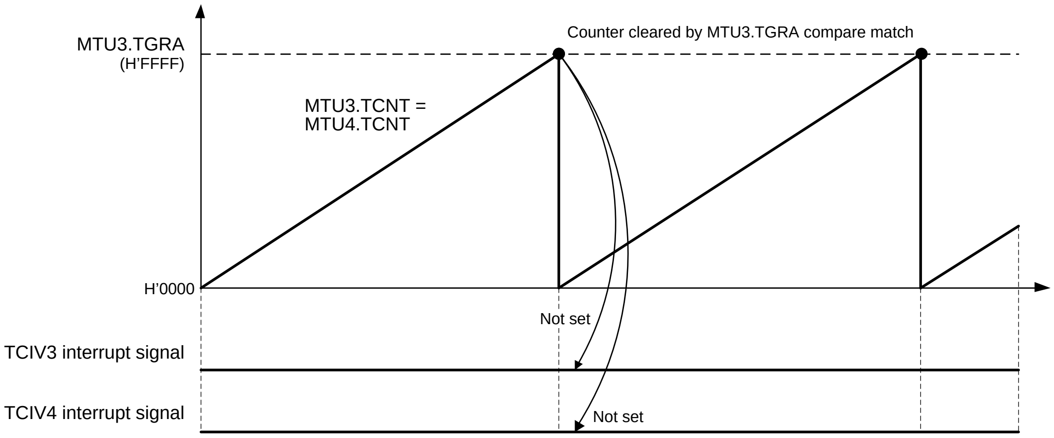


Figure 16.146 Overflow in Reset-Synchronized PWM Mode

16.6.16 Contention between Overflow/Underflow and Counter Clearing

If an overflow/underflow and counter clearing occur simultaneously, a TCIVn interrupt (n = 0 to 4 or 6 to 8) nor a TCIUn interrupt (n = 1 or 2) is not generated and TCNT clearing takes precedence.

Figure 16.147 shows the operation timing when a TGR compare match is specified as the clearing source and TGR is set to H’FFFF.

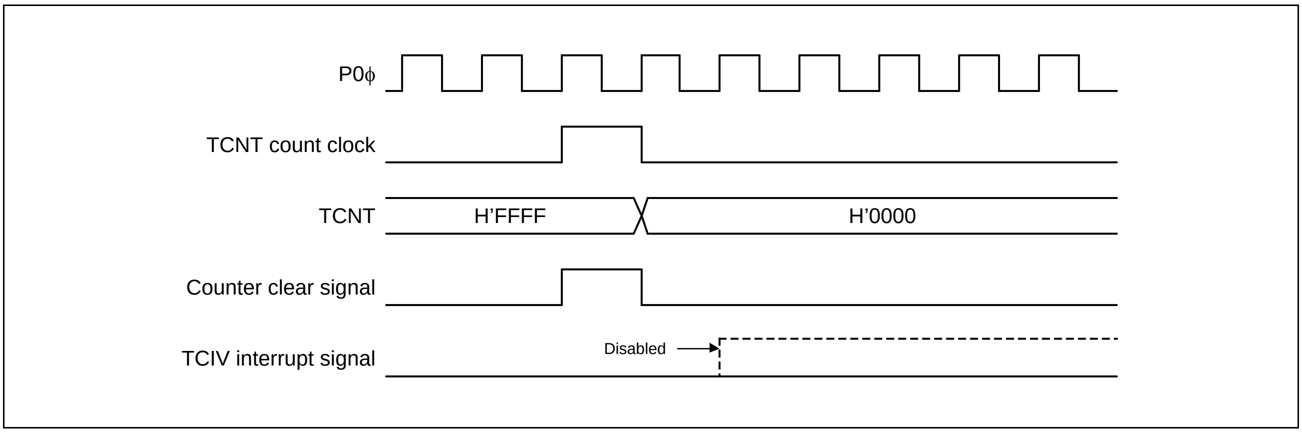


Figure 16.147 Contention between Overflow and Counter Clearing

16.6.17 Contention between TCNT Write Operation and Overflow/Underflow

If TCNT counts up or down in a TCNT write cycle and an overflow or an underflow occurs, the TCNT write operation takes precedence. A TCIVn interrupt (n = 0 to 4 or 6 to 8) nor a TCIUn interrupt (n = 1 or 2) is not generated.

Figure 16.148 shows the operation timing when there is contention between TCNT write operation and overflow.

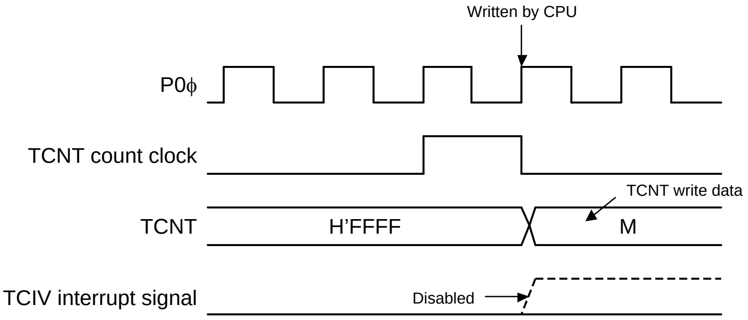


Figure 16.148 Contention between TCNT Write Operation and Overflow

16.6.18 Note on Transition from Normal Mode or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from normal mode or PWM mode 1 to reset-synchronized PWM mode in MTU3 and MTU4 (or MTU6 and MTU7), if the counter is stopped while the output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4C, MTIOC4B, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7C, MTIOC7B, and MTIOC7D) are held at a high level and then operation is started after a transition to reset-synchronized PWM mode, the initial pin output will not be correct.

When making a transition from normal mode to reset-synchronized PWM mode, write H’11 to MTU3.TIORH, MTU3.TIORL, MTU4.TIORH, and MTU4.TIORL (MTU6.TIORH, MTU6.TIORL, MTU7.TIORH, and MTU7.TIORL) to initialize the output pin state to a low level, then set the registers to the initial value (H’00) before making the mode transition.

When making a transition from PWM mode 1 to reset-synchronized PWM mode, switch to normal mode, initialize the output pin state to a low level, and then set the registers to the initial value (H’00) before making the transition to reset-synchronized PWM mode.

16.6.19 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When MTU3 and MTU4 (or MTU6 and MTU7) are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is determined by the OLSP and OLSN bits in the timer output control register (TOCR1A or TOCR1B). In complementary PWM mode or reset-synchronized PWM mode, TIOR should be set to H’00. The output level in negative phase when the TDER bit in TDERA (TDERB) is set to 0 in complementary PWM mode (the dead time is not generated) does not depend on the setting of the OLSN bit in TOCR1A (TOCR1B). It is equivalent to the inverted level of positive phase output based on the setting of the OLSP bit in TOCR1A (TOCR1B).

16.6.20 Simultaneous Input Capture in MTU1.TCNT and MTU2.TCNT in Cascade Connection

When timer counters 1 and 2 (MTU1.TCNT and MTU2.TCNT) operate as a 32-bit counter in cascade connection, the cascaded counter value cannot be captured successfully in some cases even if input-capture input is simultaneously done to MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B. This is because the input timing of MTIOC1A and MTIOC2A or to MTIOC1B and MTIOC2B may not be the same when external input-capture signals input into MTU1.TCNT and MTU2.TCNT are taken in synchronization with the internal clock.

For example, MTU1.TCNT (the counter for upper 16 bits) does not capture the count-up value by an overflow from MTU2.TCNT (the counter for lower 16 bits) but captures the count value before the up-counting. In this case, the values of MTU1.TCNT = H’FFF1 and MTU2.TCNT = H’0000 should be transferred to MTU1.TGRA and MTU2.TGRA or to MTU1.TGRB and MTU2.TGRB, but the values of MTU1.TCNT = H’FFF0 and MTU2.TCNT = H’0000 are erroneously transferred.

The MTU has a function that allows simultaneous capture of MTU1.TCNT and MTU2.TCNT with a single input capture input. This function can be used to read the 32-bit counter such that MTU1.TCNT and MTU2.TCNT are captured at the same time. For details, see **Section 16.2.11, Timer Input Capture Control Register (TICCR)**.

16.6.21 Interrupt-Skipping Function 2

When interrupt-skipping function 2 is in use and the difference between the values in MTU4.TADCORA and MTU4.TADCORB is small, correct counting of the number skipped may not be possible, in which case requests for A/D conversion will not be generated with the expected timing. The conditions listed below thus apply to these settings.

For MTU6 and MTU7, the same conditions apply to the settings of MTU7.TADCORA and MTU7.TADCORB.

(1) When the number skipped is zero for skipping function 2

- The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least four.
- The interval of comparison for MTU4.TADCORA must be at least four cycles of P0ϕ clock (the updated value of MTU4.TADCORA is set to the previous value plus or minus at least four).
- The interval of comparison for MTU4.TADCORB must be at least four cycles of P0ϕ clock (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least four).

(2) When the number skipped is one or more for skipping function 2

- The difference between the values in MTU4.TADCORA and MTU4.TADCORB must be at least two.
- The interval of comparison for MTU4.TADCORB must be at least two cycles of P0ϕ (the updated value of MTU4.TADCORB is set to the previous value plus or minus at least two).

16.6.22 Notes When Complementary PWM Mode Output Protection Function is Not Used

The complementary PWM mode output protection function is initially enabled. For details, refer to **Section 17, Port Output Enable 3 (POE3)**.

16.6.23 Notes Regarding Timer Counter (MTU5.TCNT) and Timer General Register (MTU5.TGR)

Do not set TGRj (j = U, V, or W) in MTU5 to the value of the corresponding TCNTj in MTU5 plus one while counting by TCNTj in MTU5 is stopped. If such a setting is made, a compare match will be generated even though counting is stopped.

In this case, if the value of the compare match enable bit (TGIE5j bit of TIER in MTU5) is 1 (enabled), a compare match interrupt will also be generated. If the value of the timer compare match clear register is 1 (enabled), the TCNTj counter in MTU5 is automatically cleared to H’0000 when the compare match is generated, regardless of whether compare match interrupts are enabled or disabled.

16.6.24 Notes to Prevent Malfunctions in Synchronous Clearing for Complementary PWM Mode

If control of the output waveform is enabled (the WRE bit of TWCRA or TWCRB is 1) at the time of synchronous counter clearing in complementary PWM mode, satisfaction of either condition 1 or 2 below has the following effects.

- Dead time on the PWM output pins is shortened (or disappears).
- The active level is output on the negative phase PWM output pins beyond the period for active-level output.

Condition 1:

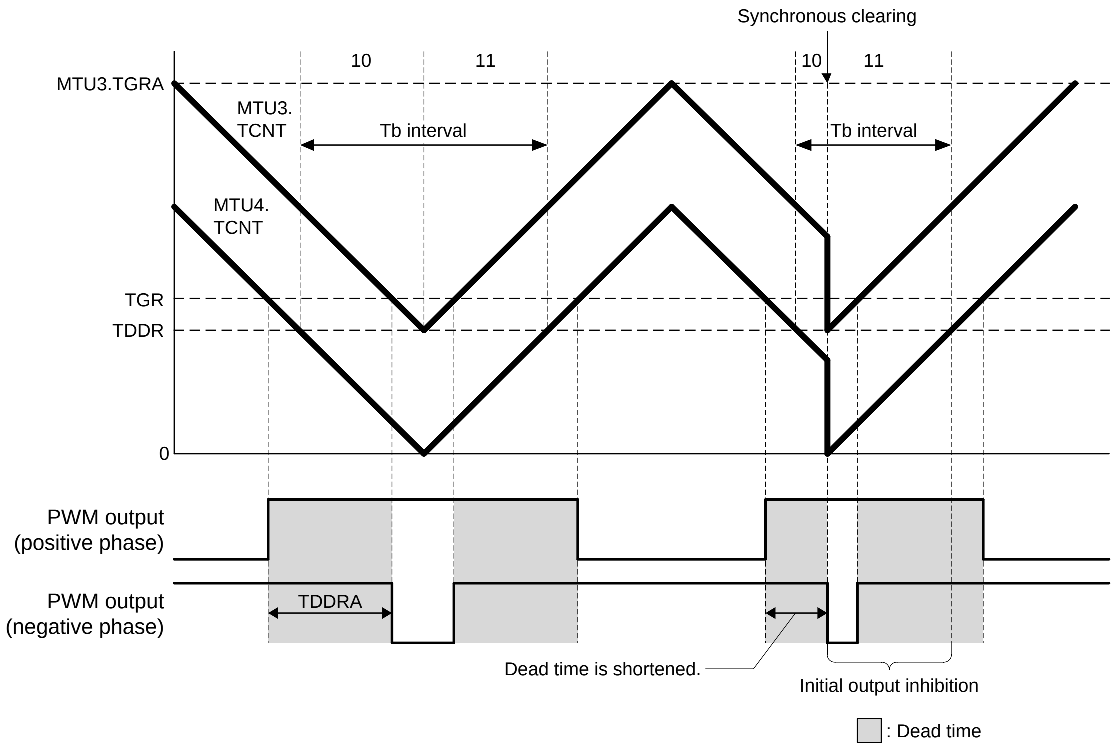
In portion (10) of the initial output inhibition period in **Figure 16.149**, synchronous clearing occurs within the dead-time period for PWM output.

Condition 2:

In portions (10) and (11) of the initial output inhibition period in **Figure 16.150**, synchronous clearing occurs when any condition from among TGRB in MTU3 (MTU6) ≤ TDDR (TDDRB), TGRA in MTU4 (MTU7) ≤ TDDR (TDDRB), and TGRB in MTU4 (MTU7) ≤ TDDR (TDDRB) is satisfied.

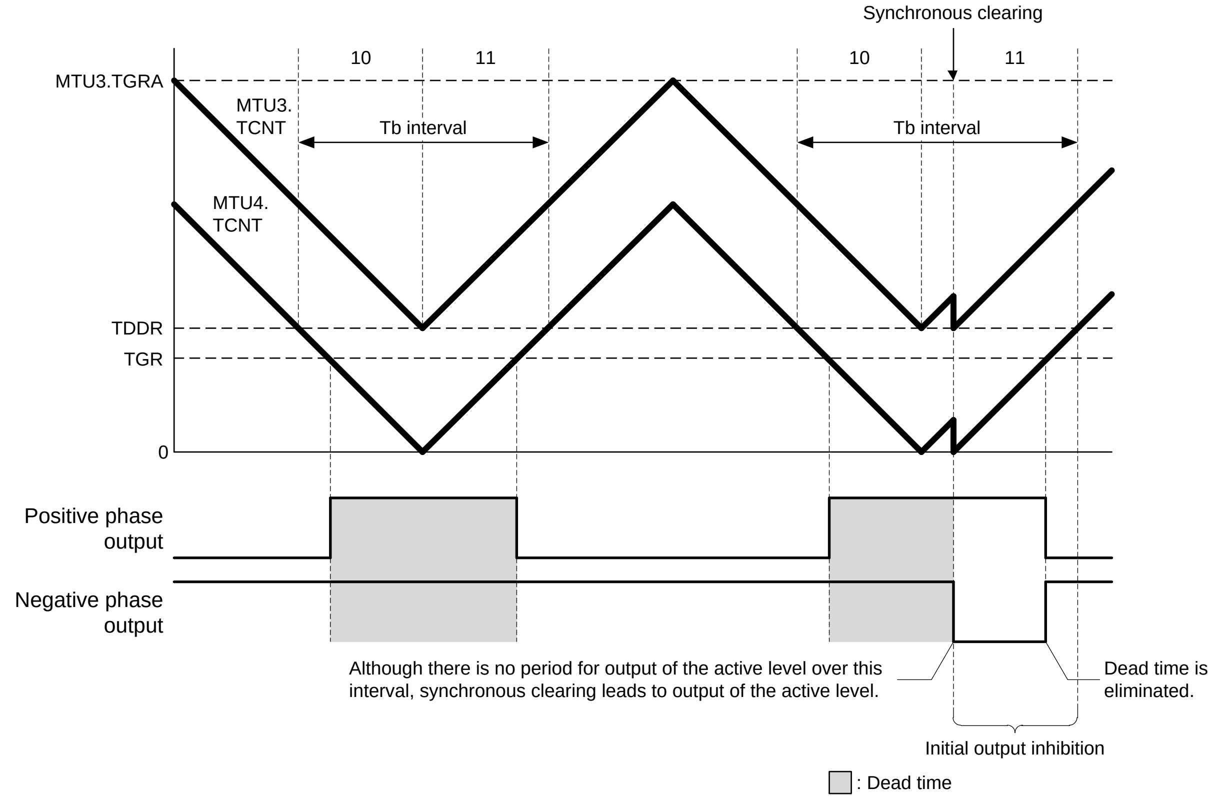
The following method avoids the above phenomena.

Ensure that synchronous clearing proceeds with the value of each comparison register (TGRB in MTU3 (MTU6), TGRA in MTU4 (MTU7), and TGRB in MTU4 (MTU7)) set to at least double the value of the TDDRA register (TDDRB register).



Note: PWM output is active low.

Figure 16.149 Example of Synchronous Clearing (When Condition 1 Applies)



Note: PWM output is active low.

Figure 16.150 Example of Synchronous Clearing (When Condition 2 Applies)

16.6.25 Continuous Output of Interrupt Signal in Response to a Compare Match

When the TGR register is set to H’0000, the P0ϕ/1 clock is set as the count clock, and compare match is set as the trigger for clearing of the count clock, the value of the TCNT counter remains H’0000, and the interrupt signal will be output continuously (i.e. its level will be flat) rather than output over a single cycle. Consequently, interrupts will not be detected in response to second and subsequent compare matches.

Figure 16.151 shows the timing for continuous output of the interrupt signal in response to a compare match.

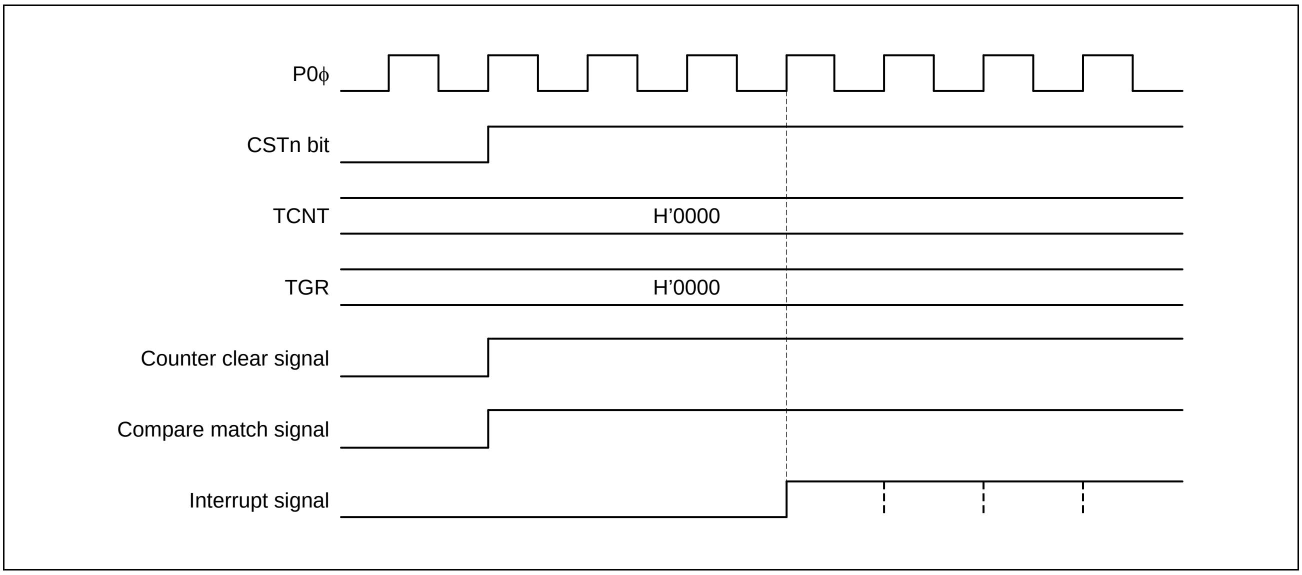
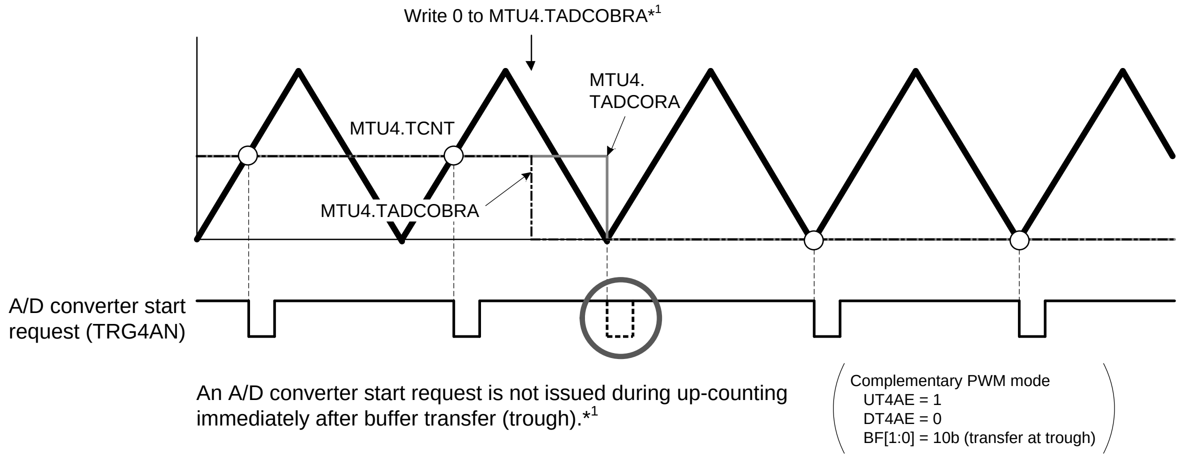


Figure 16.151 Continuous Output of Interrupt Signal in Response to a Compare Match

16.6.26 Notes on A/D Converter Start Request Delaying Function in Complementary PWM Mode

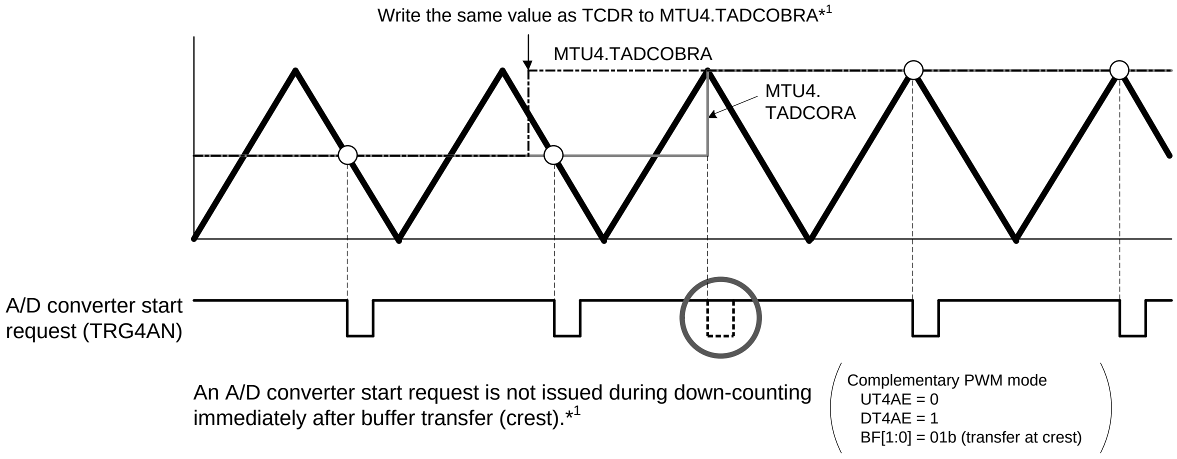
- When the TADCOBRA or TADCOBRB register in MTU4 (MTU7) is set to 0, the UT4AE or UT4BE (UT7AE or UT7BE) bit of TADCR in MTU4 (MTU7) is set to 1, and buffer transfer is performed at the trough of the value of the TCNT counter in MTU4 (MTU7), an A/D converter start request is not issued during up-counting immediately after the transfer (**Figure 16.152**).
- When the TADCOBRA or TADCOBRB register in MTU4 (MTU7) is set to the same value as that in the TCDR register, the DT4AE or DT4BE (DT7AE or DT7BE) bit of TADCR in MTU4 (MTU7) is set to 1, and buffer transfer is performed at the crest of the value of the TCNT counter in MTU4 (MTU7), an A/D converter start request is not issued during down-counting immediately after the transfer (**Figure 16.153**).
- When using A/D converter start requests in coordination with the interrupt skipping function, set up the TADCORA or TADCORB register in MTU4 (MTU7) so that the setting satisfies the condition "2 ≤ TADCORA or TADCORB in MTUn ≤ TCDR - 2" (n = 4 or 7).



Note: UT4AE, DT4AE, BF[1:0]: Bits in TADCR

Note 1. An A/D converter start request is issued when TCDR − 1 ≥ MTU4.TADCOBRA/TADCOBRB ≥ 1 is written.

Figure 16.152 A/D Converter Start Request when TADCOBRA is Set to 0 in MTU4



An A/D converter start request is not issued during down-counting immediately after buffer transfer (crest).*1

Note: UT4AE, DT4AE, BF[1:0]: Bits in TADCR

Note 1. An A/D converter start request is issued when TCDR −1 ≥ MTU4.TADCOBRA/TADCOBRB ≥ 1 is written.

Figure 16.153 A/D Converter Start Request when TADCOBRA is Set to the Same Value as TCDR in MTU4

16.7 MTU Output Pin Initialization

16.7.1 Operating Modes

The MTU has the following six operating modes. Waveforms can be output in any of these modes.

- Normal mode (MTU0 to MTU4 and MTU6 to MTU8)
- PWM mode 1 (MTU0 to MTU4, MTU6, and MTU)
- PWM mode 2 (MTU0 to MTU2)
- Phase counting modes 1 to 5 (MTU1 and MTU2)
- Complementary PWM mode (MTU3, MTU4, MTU6, and MTU7)
- Reset-synchronized PWM mode (MTU3, MTU4, MTU6, and MTU7)

This section describes how to initialize the MTU output pins in each of these modes.

16.7.2 Operation in Case of Re-Setting Due to Error during Operation

If an error occurs during MTU operation, MTU output should be cut off by the system. The output can be cut off by allowing non-active level output from the pins by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports. MTU output can be disabled through TIOR settings. Complementary PWM output (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D, MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D) should be specified through TOERA and TOERB settings. For PWM output pins, output can also be cut by hardware, using port output enable 3 (POE3). The pin initialization procedures for re-setting due to an error during operation and the procedures for restarting in a different mode after re-setting are described below.

The MTU has six operating modes, as stated above. There are thus 36 mode transition combinations, but some transitions are not available with certain channel and mode combinations. Available mode transition combinations are shown in **Table 16.80**.

Table 16.80 Mode Transition Combinations

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	Not available	Not available
PCM	(17)	(18)	(19)	(20)	Not available	Not available
CPWM	(21)	(22)	Not available	Not available	(23) (24)	(25)
RPWM	(26)	(27)	Not available	Not available	(28)	(29)

Note: Normal: Normal mode
PWM1: PWM mode 1
PWM2: PWM mode 2
PCM: Phase counting modes 1 to 5
CPWM: Complementary PWM mode
RPWM: Reset-synchronized PWM mode

16.7.3 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation

- When making a transition to a mode (Normal, PWM1, PWM2, or PCM) in which the pin output level is selected by the timer I/O control register (TIOR) setting, initialize the pins by means of TIOR setting.
- In PWM mode 1, waveforms are not output to the MTIOCnB and MTIOCnD pins (n = 3, 4, 6, or 7). If the MTIOCnB or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers. If the MTIOCnm output (n = 0 to 2 and m = A to D) is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- In normal mode or PWM mode 2, if TGRC and TGRD operate as buffer registers, waveforms are not output to the corresponding MTIOCnC and MTIOCnD pins (n = 0, 3, 4, 6, or 7). If the MTIOCnC or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- In PWM mode 1, if either TGRC or TGRD operates as a buffer register, waveforms are not output to the corresponding MTIOCnC and MTIOCnD pins (n = 0, 3, 4, 6, or 7). If the MTIOCnC or MTIOCnD output is selected as the function of the pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.
- When making a transition to a mode (CPWM or RPWM) in which the pin output level is selected by the timer output control register (TOCR1A, TOCR2A, TOCR1B, or TOCR2B) setting, temporarily disable output in MTU3 and MTU4 (or MTU6 and MTU7) with the timer output master enable register (TOERA or TOERB). If the MTIOCnm output (n =3, 4, 6, or 7 and m = A to D) is selected as the function of pins, they enter high-impedance state. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports. Switch to normal mode, perform initialization with TIOR, restore TIOR to its initial value, then operate the MTU in accordance with the mode setting procedure (TOCR1A setting, TOCR2A setting, TMDR1 setting, and TOERA setting (TOCR1B setting, TOCR2B setting, TMDR1 setting, and TOERB setting)).

NOTE

Channel number is substituted for “n” indicated in this section.

Pin initialization procedures are described below for the numbered combinations in **Table 16.80**. The active level is assumed to be low.

(1) Operation When Error Occurs in Normal Mode and Operation is Restarted in Normal Mode

Figure 16.154 shows a case in which an error occurs in normal mode and operation is restarted in normal mode after re- setting.

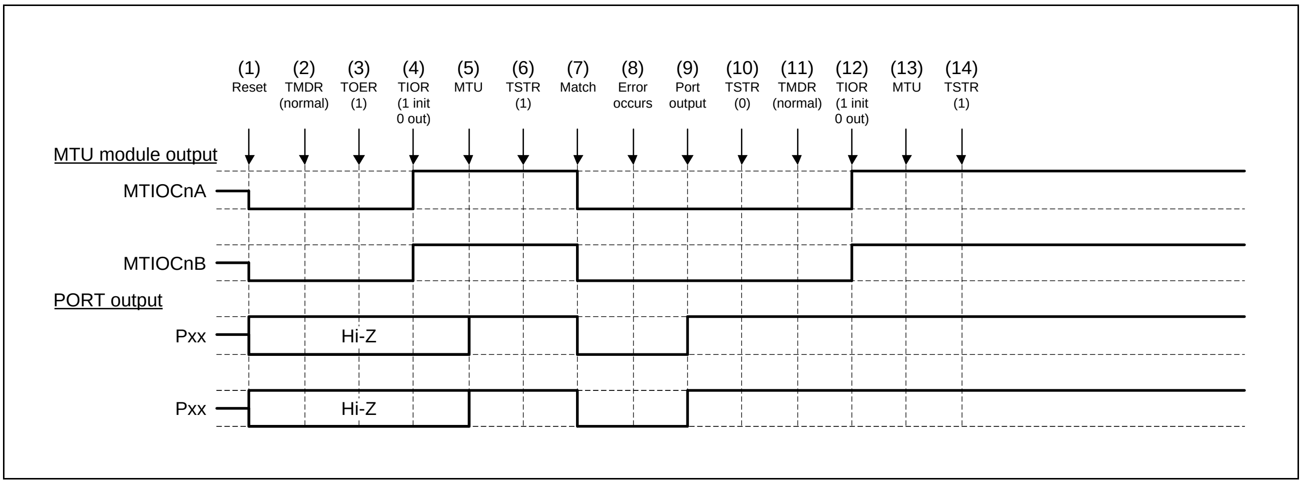


Figure 16.154 Error Occurrence in Normal Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) After a reset, the TMDR1 setting is for normal mode.

(3) For MTU3 and MTU4 (MTU6 and MTU7), enable output with TOERA (TOERB) before initializing the pins with TIOR.

(4) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)

(5) Set MTU output using the port mode control registers (PMCn) and port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) Output goes low on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB).

(11) This step is not necessary when restarting in normal mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(2) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM Mode 1

Figure 16.155 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.

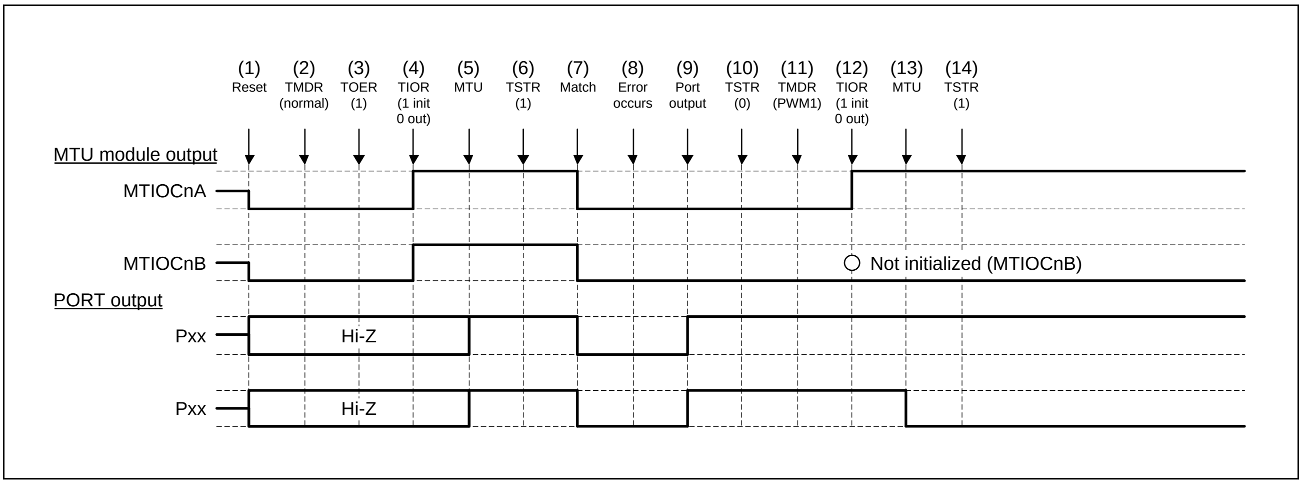


Figure 16.155 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.154**.

(11) Set PWM mode 1.

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(3) Operation When Error Occurs in Normal Mode and Operation is Restarted in PWM mode 2

Figure 16.156 shows a case in which an error occurs in normal mode and operation is restarted in PWM mode 2 after re-setting.

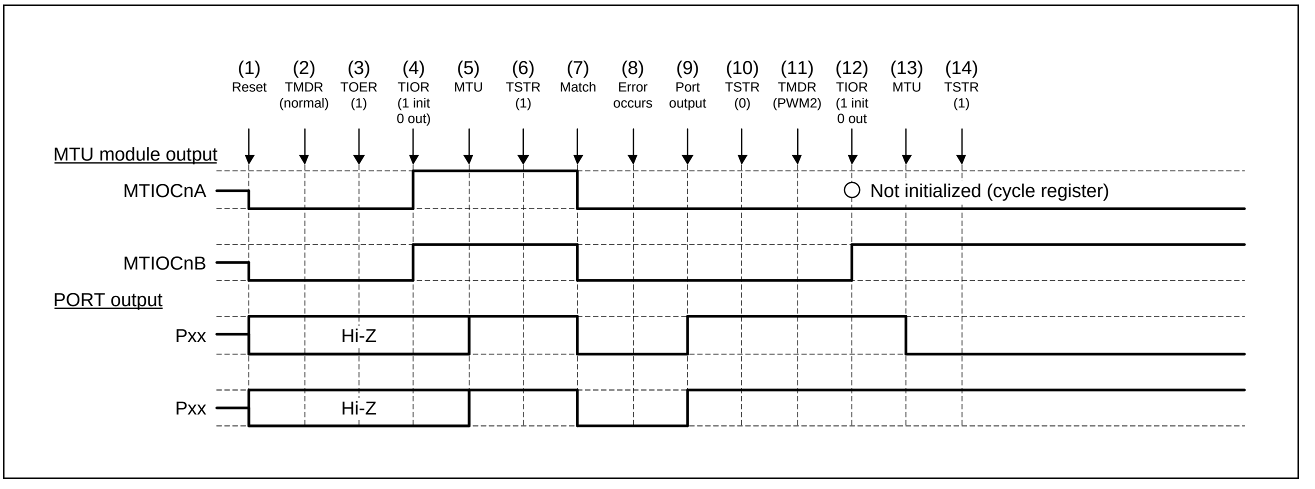


Figure 16.156 Error Occurrence in Normal Mode, Recovery in PWM Mode 2

(1) to (10) are the same as in **Figure 16.154**.

(11) Set PWM mode 2.

(12) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers. To output a specified level, set up the general output ports with the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

PWM mode 2 can only be selected for MTU0 to MTU2, and therefore TOERA setting is not necessary.

(4) Operation When Error Occurs in Normal Mode and Operation is Restarted in Phase Counting Mode

Figure 16.157 shows a case in which an error occurs in normal mode and operation is restarted in phase counting mode after re-setting.

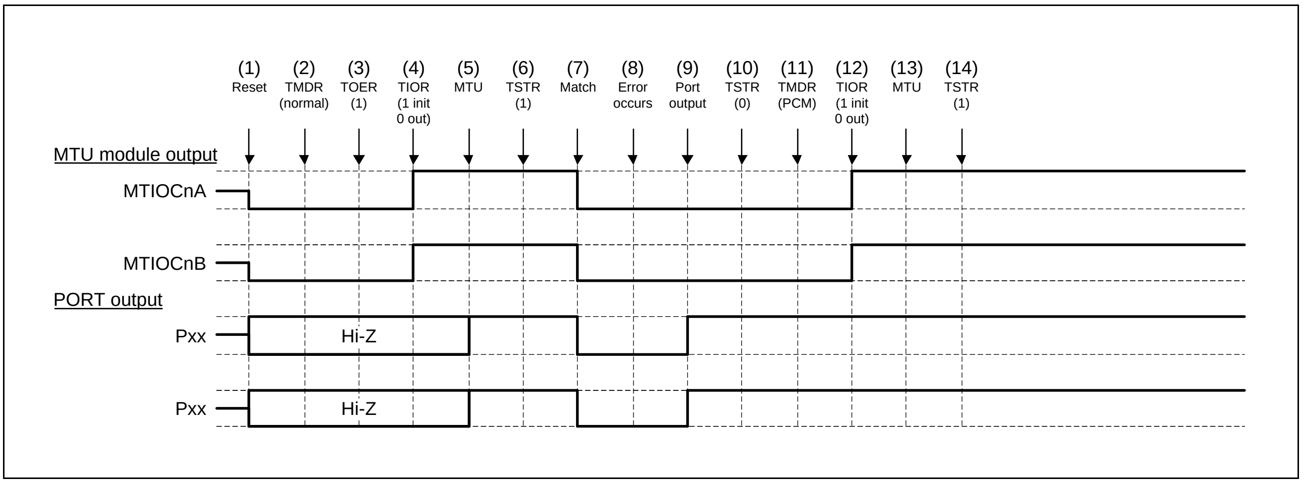


Figure 16.157 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode

(1) to (10) are the same as in **Figure 16.154**.

(11) Set the phase counting mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

The phase counting mode can only be selected for MTU1 and MTU2, and therefore TOERA setting is not necessary.

(5) Operation When Error Occurs in Normal Mode and Operation is Restarted in Complementary PWM Mode

Figure 16.158 shows a case in which an error occurs in normal mode and operation is restarted in complementary PWM mode after re-setting.

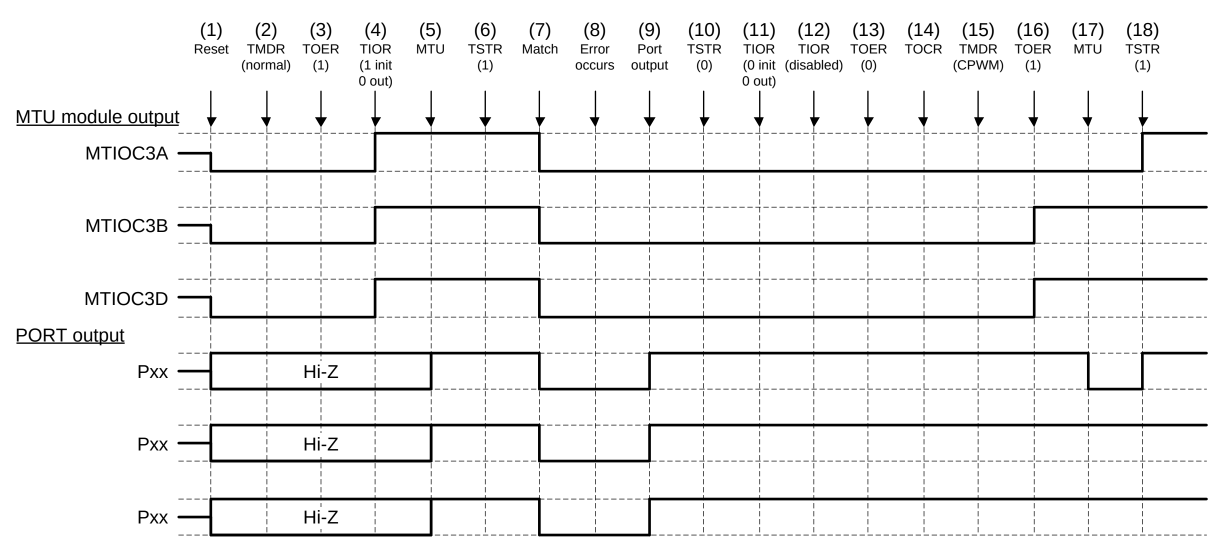


Figure 16.158 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in **Figure 16.154**.

(11) Initialize the normal mode waveform generation section with TIOR.

(12) Disable operation of the normal mode waveform generation section with TIOR.

(13) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(14) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(15) Set complementary PWM mode.

(16) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(17) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(18) Restart operation by setting TSTRA (TSTRB).

(6) Operation When Error Occurs in Normal Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.159 shows a case in which an error occurs in normal mode and operation is restarted in reset-synchronized PWM mode after re-setting.

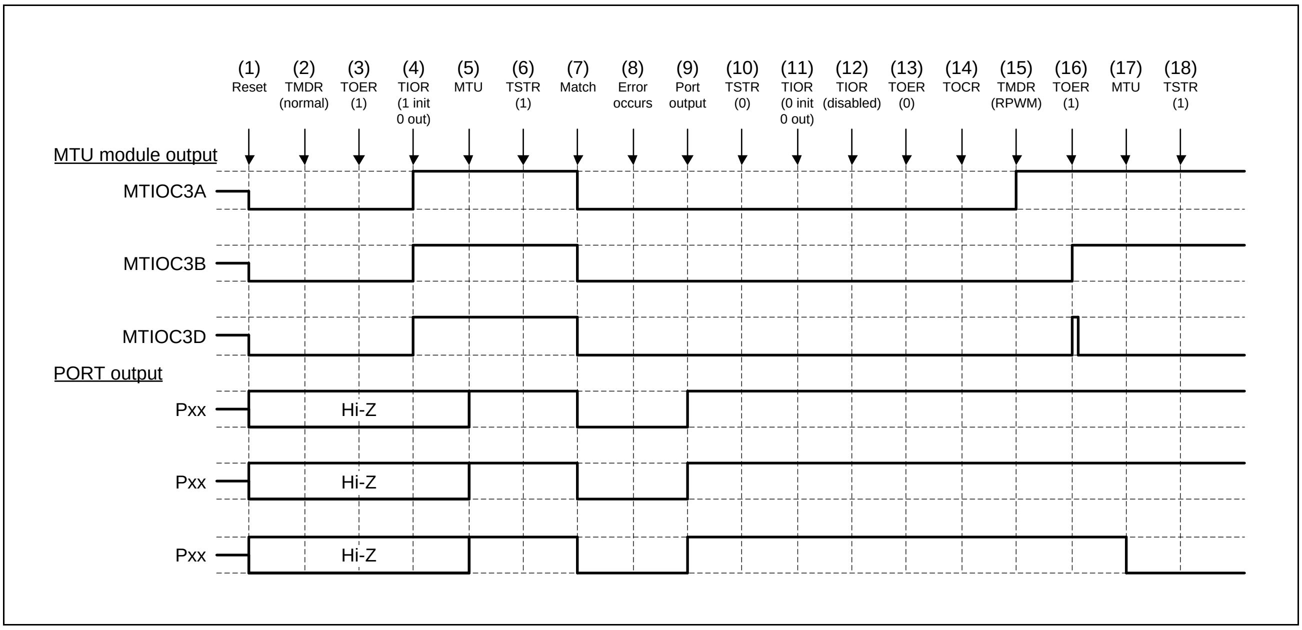


Figure 16.159 Error Occurrence in Normal Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (13) are the same as in **Figure 16.156**.

(14) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(15) Set reset-synchronized PWM mode.

(16) Enable output from MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(17) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(18) Restart operation by setting TSTRA (TSTRB).

(7) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Normal Mode

Figure 16.160 shows a case in which an error occurs in PWM mode 1 and operation is restarted in normal mode after re-setting.

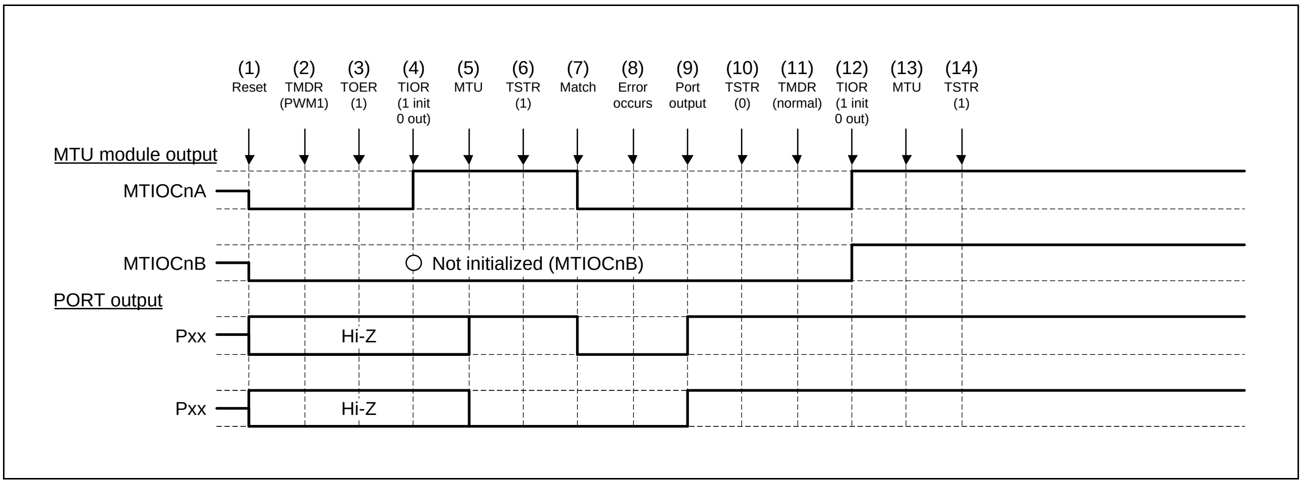


Figure 16.160 Error Occurrence in PWM Mode 1, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Set PWM mode 1.

(3) For MTU3 and MTU4 (MTU6 and MTU7), enable output with TOERA (TOERB) before initializing the pins with TIOR.

(4) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 1, the MTIOCnB side is not initialized.)

(5) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) Output goes low on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB).

(11) Set normal mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(8) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 1

Figure 16.161 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 1 after re-setting.

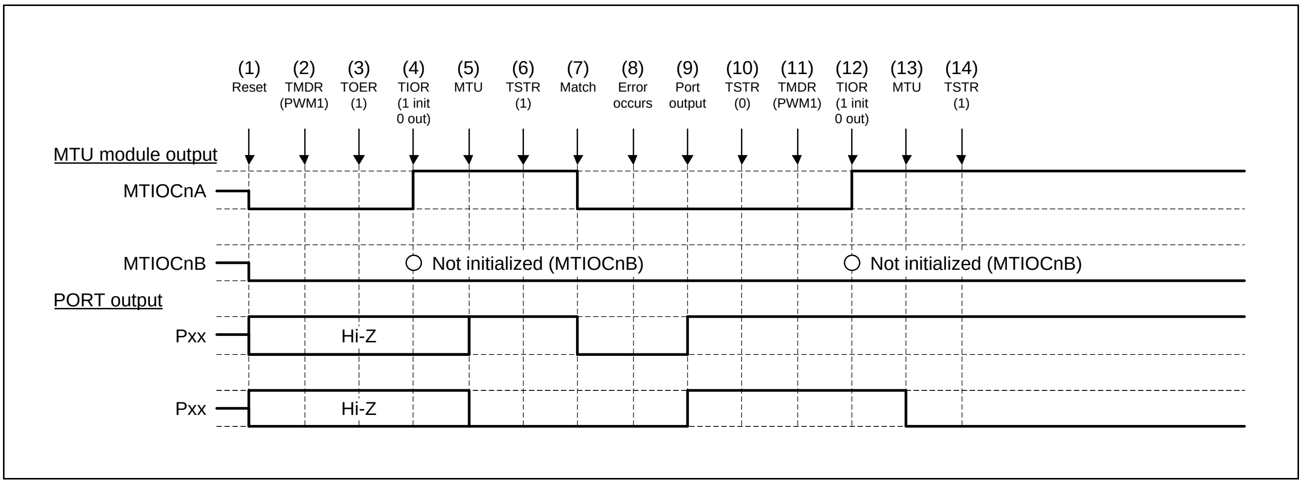


Figure 16.161 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.160**.

(11) This step is not necessary when restarting in PWM mode 1.

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(9) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in PWM mode 2

Figure 16.162 shows a case in which an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.

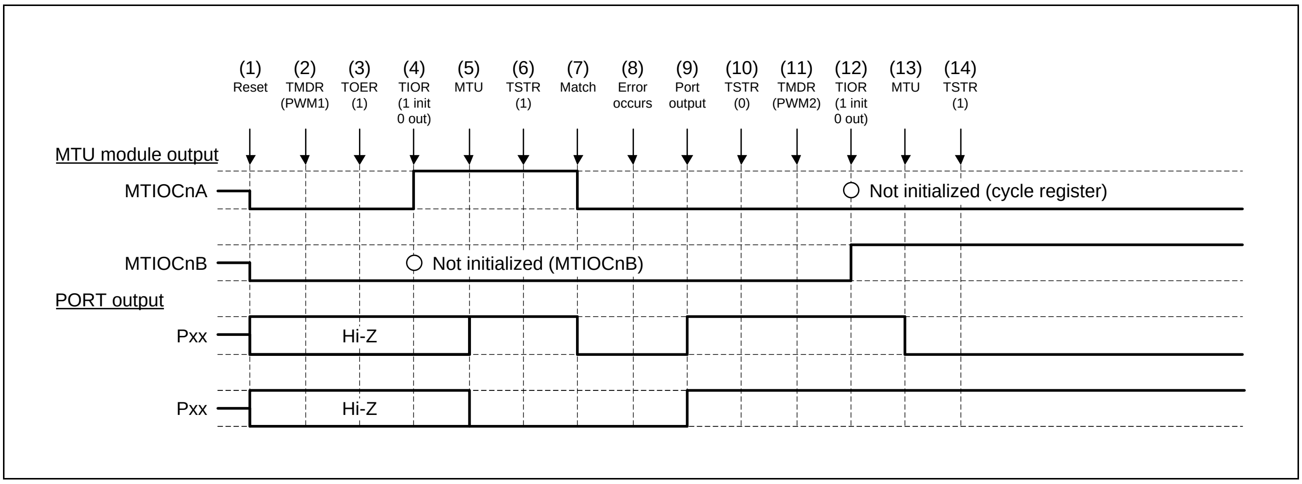


Figure 16.162 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

(1) to (10) are the same as in **Figure 16.160**.

(11) Set PWM mode 2.

(12) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

PWM mode 2 can only be selected for MTU0 to MTU2, and therefore TOERA setting is not necessary.

(10) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Phase Counting Mode

Figure 16.163 shows a case in which an error occurs in PWM mode 1 and operation is restarted in phase counting mode after re-setting.

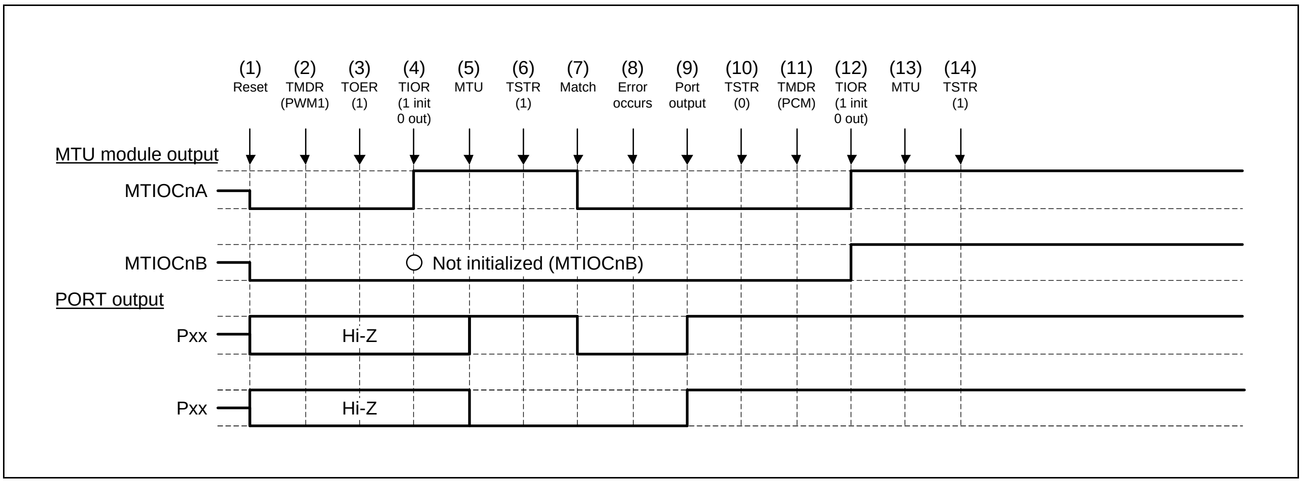


Figure 16.163 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

(1) to (10) are the same as in **Figure 16.160**.

(11) Set the phase counting mode.

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA.

NOTE

The phase counting mode can only be selected for MTU1 and MTU2, and therefore TOERA setting is not necessary.

(11) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Complementary PWM Mode

Figure 16.164 shows a case in which an error occurs in PWM mode 1 and operation is restarted in complementary PWM mode after re-setting.

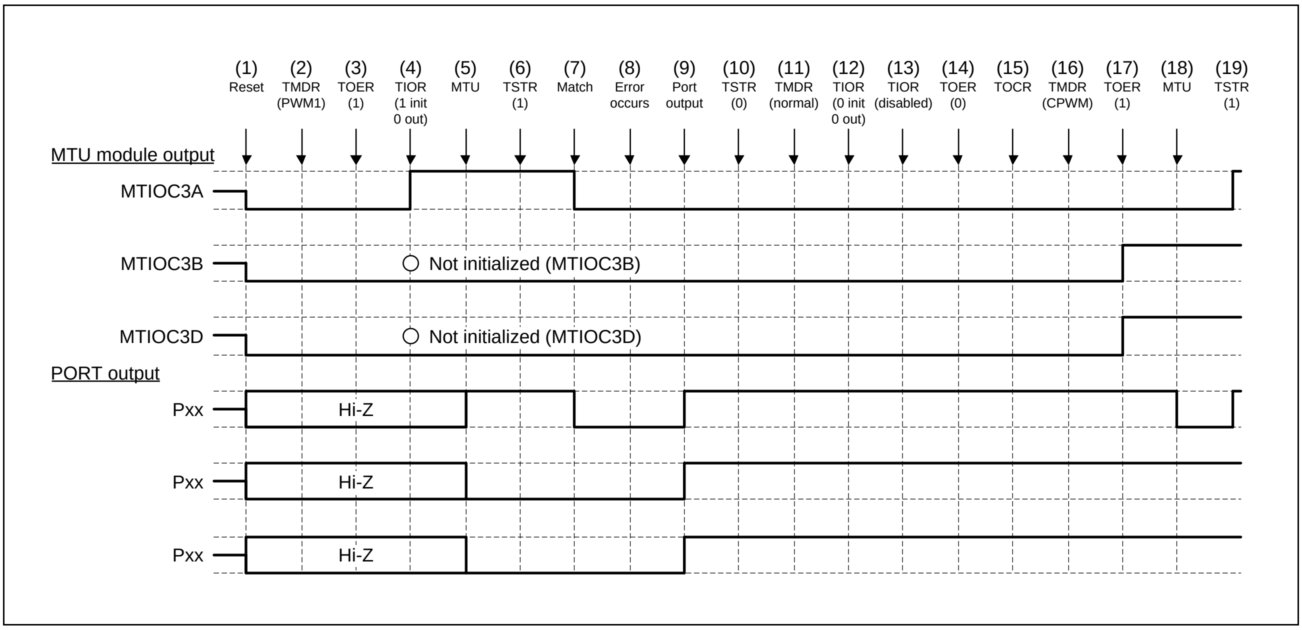


Figure 16.164 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

(1) to (10) are the same as in **Figure 16.160**.

(11) Set normal mode to initialize the normal mode waveform generation section.

(12) Initialize the PWM mode 1 waveform generation section with TIOR.

(13) Disable operation of the PWM mode 1 waveform generation section with TIOR

(14) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(15) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(16) Set complementary PWM mode.

(17) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(18) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(19) Restart operation by setting TSTRA (TSTRB).

(12) Operation When Error Occurs in PWM Mode 1 and Operation is Restarted in Reset- Synchronized PWM Mode

Figure 16.165 shows a case in which an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

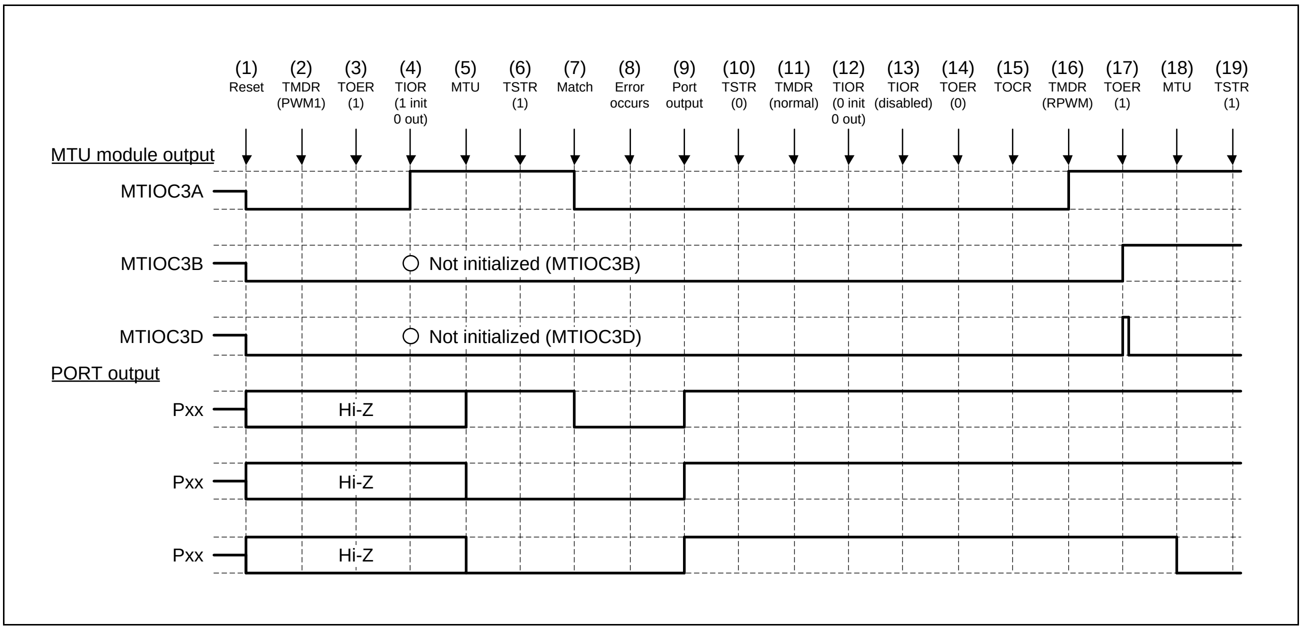


Figure 16.165 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

(1) to (14) are the same as in **Figure 16.164**.

(15) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(16) Set reset-synchronized PWM mode.

(17) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(18) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(19) Restart operation by setting TSTRA (TSTRB).

(13) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Normal Mode

Figure 16.166 shows a case in which an error occurs in PWM mode 2 and operation is restarted in normal mode after re-setting.

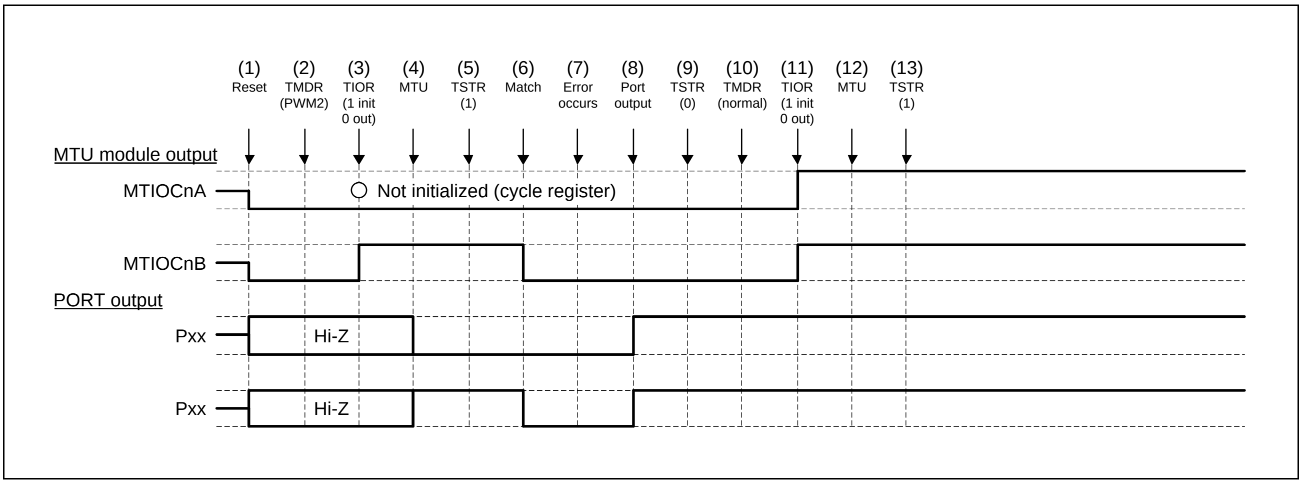


Figure 16.166 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Set PWM mode 2.

(3) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence. In PWM mode 2, the pins corresponding to the TGR registers used as cycle registers are not initialized. In the example, TGRA in MTUn is used as a cycle register.)

(4) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(5) Start count operation by setting TSTRA.

(6) Output goes low on compare match occurrence.

(7) An error occurs.

(8) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(9) Stop count operation by setting TSTRA.

(10) Set normal mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(14) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 1

Figure 16.167 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 1 after re-setting.

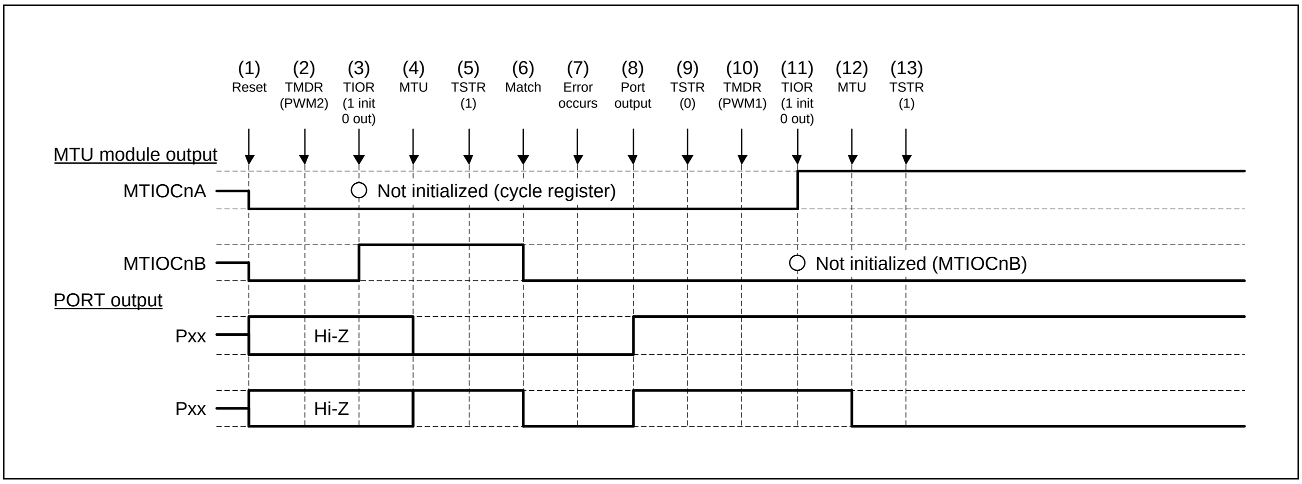


Figure 16.167 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

(1) to (9) are the same as in **Figure 16.166**.

(10) Set PWM mode 1.

(11) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(15) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in PWM Mode 2

Figure 16.168 shows a case in which an error occurs in PWM mode 2 and operation is restarted in PWM mode 2 after re-setting.

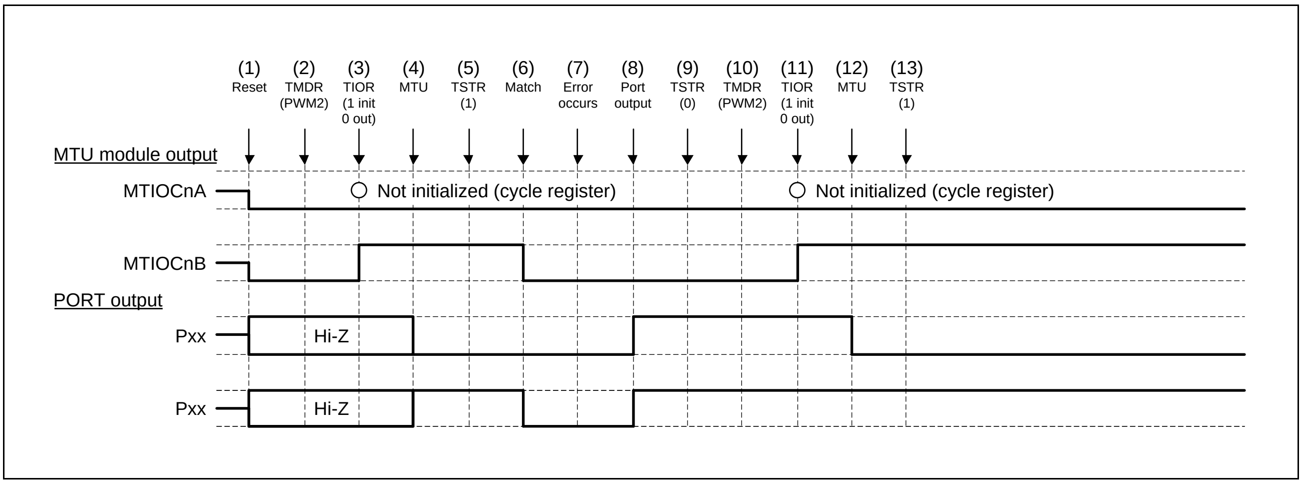


Figure 16.168 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

(1) to (9) are the same as in **Figure 16.166**.

(10) This step is not necessary when restarting in PWM mode 2.

(11) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(16) Operation When Error Occurs in PWM Mode 2 and Operation is Restarted in Phase Counting Mode

Figure 16.169 shows a case in which an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.

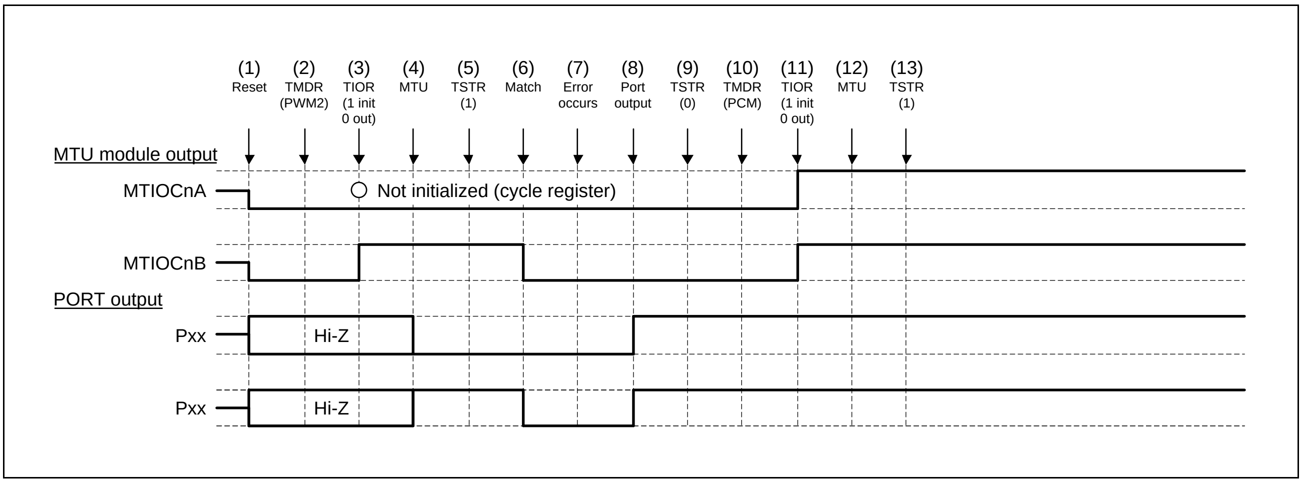


Figure 16.169 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

(1) to (9) are the same as in **Figure 16.166**.

(10) Set the phase counting mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(17) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Normal Mode

Figure 16.170 shows a case in which an error occurs in phase counting mode and operation is restarted in normal mode after re-setting.

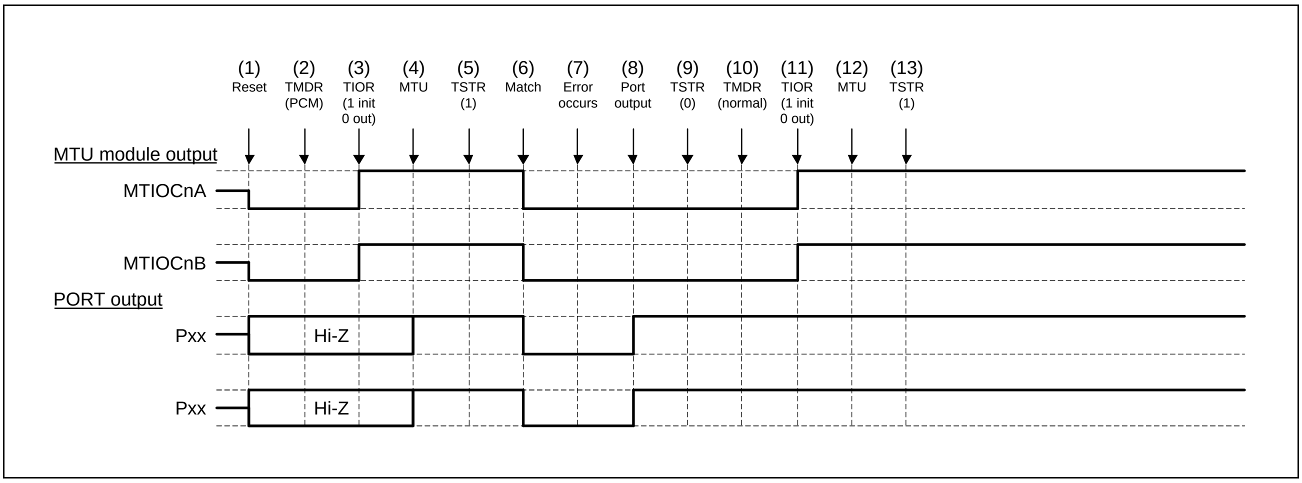


Figure 16.170 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Set phase counting mode.

(3) Initialize the pins with TIOR. (In the example, the initial output is a high level, and a low level is output on compare match occurrence.)

(4) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(5) Start count operation by setting TSTRA.

(6) Output goes low on compare match occurrence.

(7) An error occurs.

(8) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(9) Stop count operation by setting TSTRA.

(10) Set normal mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(18) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 1

Figure 16.171 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 1 after re-setting.

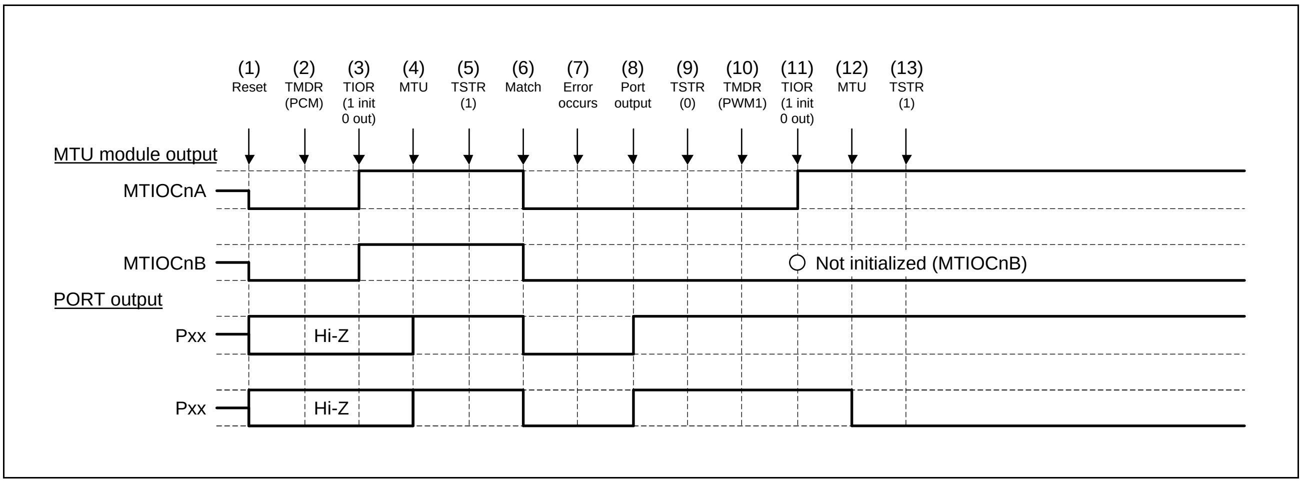


Figure 16.171 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

(1) to (9) are the same as in **Figure 16.170**.

(10) Set PWM mode 1.

(11) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(19) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in PWM Mode 2

Figure 16.172 shows a case in which an error occurs in phase counting mode and operation is restarted in PWM mode 2 after re-setting.

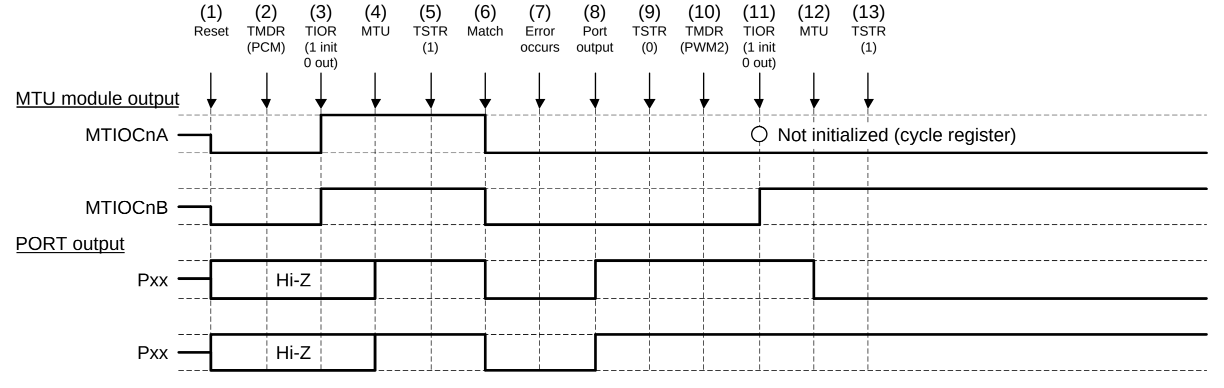


Figure 16.172 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

(1) to (9) are the same as in **Figure 16.170**.

(10) Set PWM mode 2.

(11) Initialize the pins with TIOR. (In PWM mode 2, waveforms are not output to the pins corresponding to the TGR registers used as cycle registers.
To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(20) Operation When Error Occurs in Phase Counting Mode and Operation is Restarted in Phase Counting Mode

Figure 16.173 shows a case in which an error occurs in phase counting mode and operation is restarted in phase counting mode after re-setting.

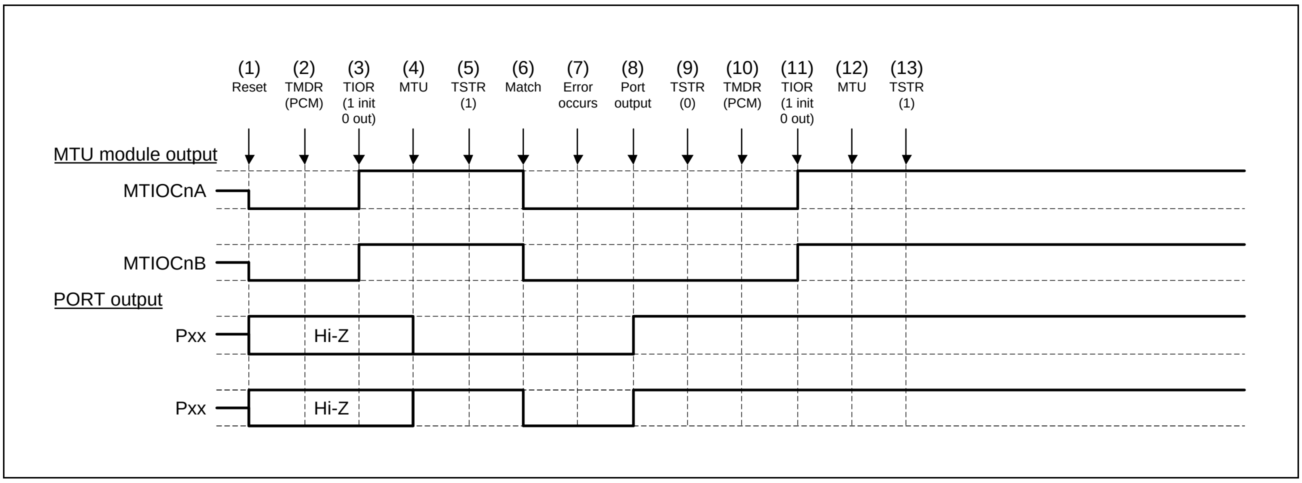


Figure 16.173 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

(1) to (9) are the same as in **Figure 16.170**.

(10) This step is not necessary when restarting in phase counting mode.

(11) Initialize the pins with TIOR.

(12) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(13) Restart operation by setting TSTRA.

(21) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Normal Mode

Figure 16.174 shows a case in which an error occurs in complementary PWM mode and operation is restarted in normal mode after re-setting.

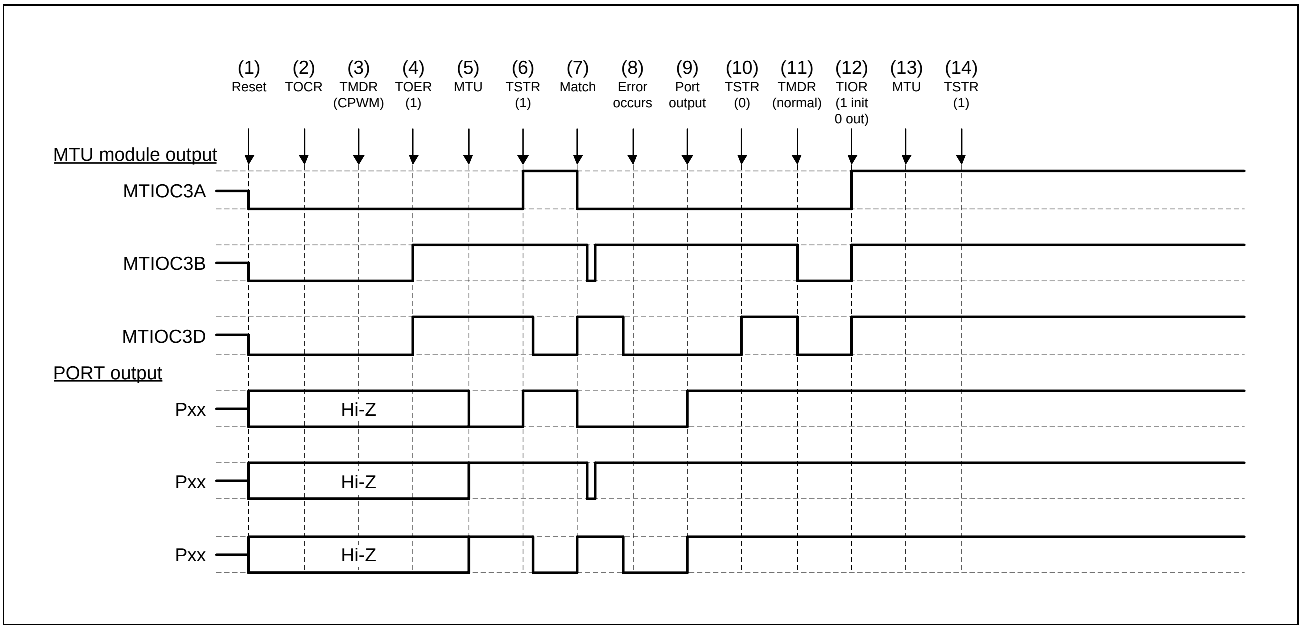


Figure 16.174 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

(1) After a reset, the MTU3 output goes low and the ports enter high-impedance state.

(2) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(3) Set complementary PWM mode.

(4) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(5) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) The complementary PWM waveform is output on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB). (MTU output becomes the initial complementary PWM output value).

(11) Set normal mode (MTU output goes low).

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(22) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in PWM Mode 1

Figure 16.175 shows a case in which an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.

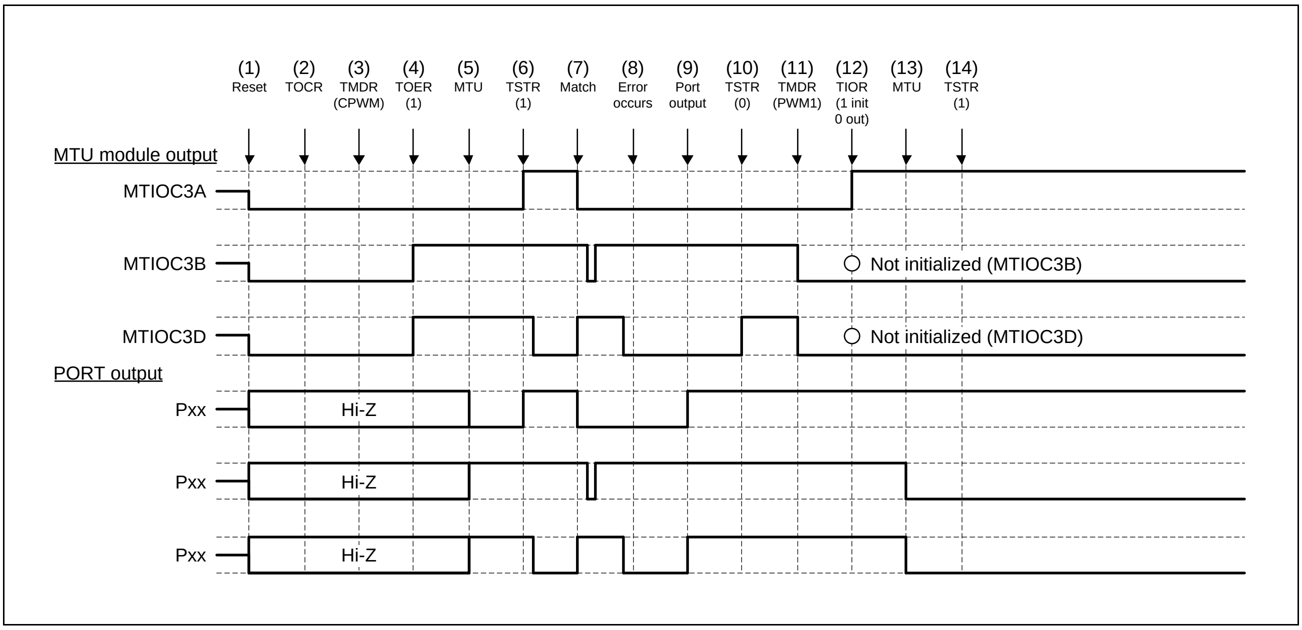


Figure 16.175 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.174**.

(11) Set PWM mode 1 (MTU output goes low).

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(23) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 16.176 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using the cycle and duty settings at the time of stopping the counter).

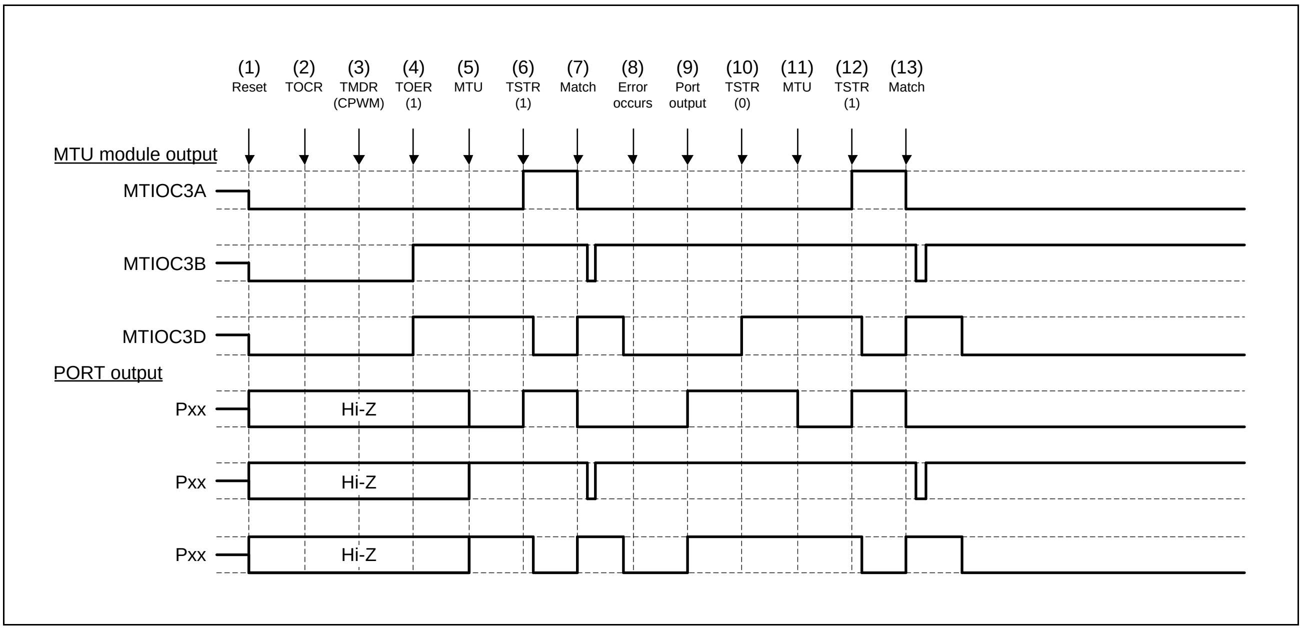


Figure 16.176 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (1)

(1) to (10) are the same as in **Figure 16.174**.

(11) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(12) Restart operation by setting TSTRA (TSTRB).

(13) The complementary PWM waveform is output on compare match occurrence.

(24) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Complementary PWM Mode with New Settings

Figure 16.177 shows a case in which an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (operation is restarted using new cycle and duty ratio settings).

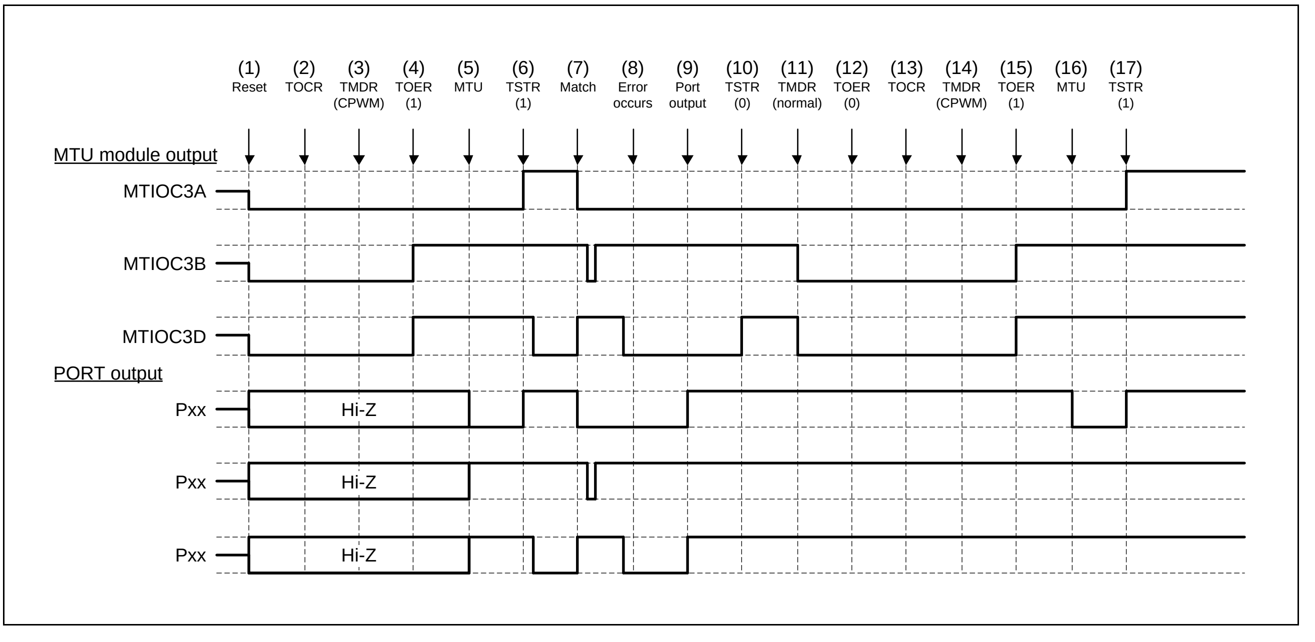


Figure 16.177 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode (2)

(1) to (10) are the same as in **Figure 16.174**.

(11) Set normal mode and make new settings (MTU output goes low).

(12) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(13) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(14) Set complementary PWM mode.

(15) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(16) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(17) Restart operation by setting TSTRA (TSTRB).

(25) Operation When Error Occurs in Complementary PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.178 shows a case in which an error occurs in complementary PWM mode and operation is restarted in reset- synchronized PWM mode after re-setting.

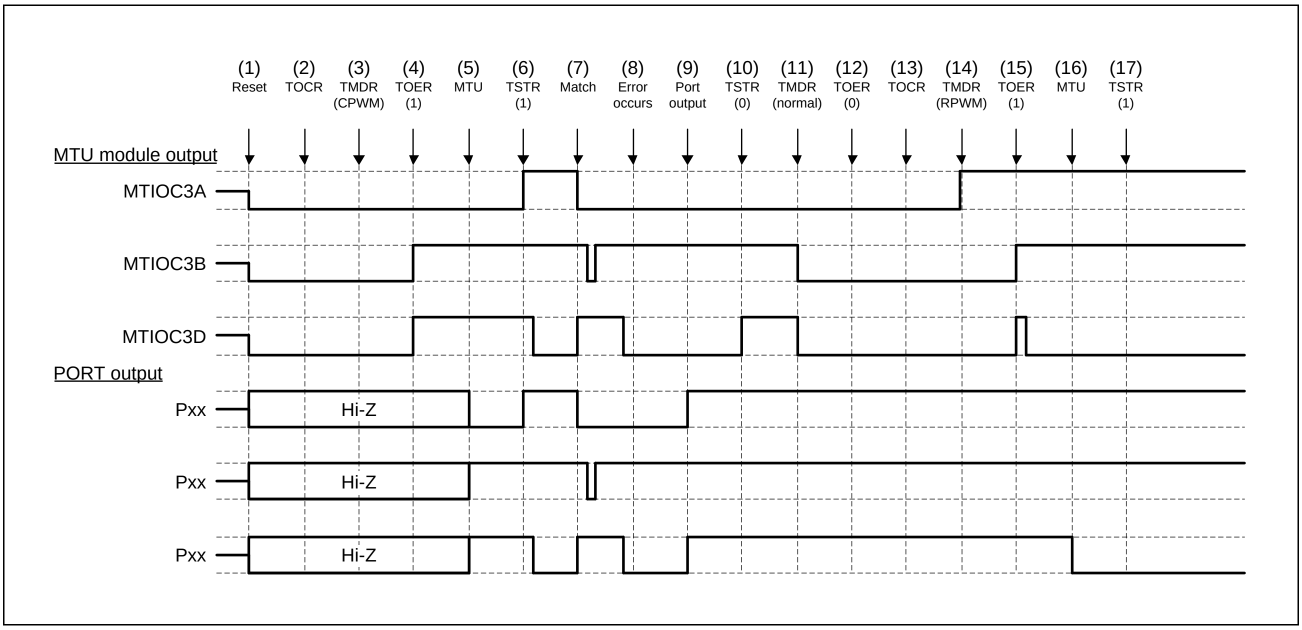


Figure 16.178 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in **Figure 16.174**.

(11) Set normal mode (MTU output goes low).

(12) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(13) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(14) Set reset-synchronized PWM mode.

(15) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(16) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(17) Restart operation by setting TSTRA (TSTRB).

(26) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Normal Mode

Figure 16.179 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in normal mode after re-setting.

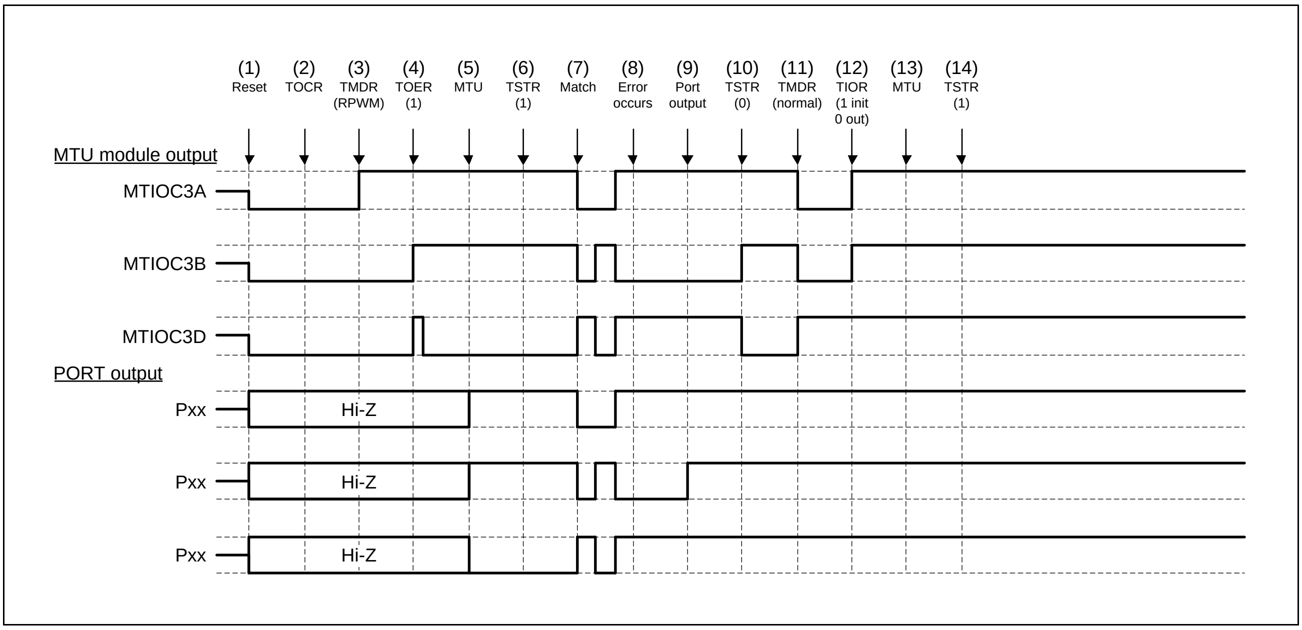


Figure 16.179 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode

(1) After a reset, the MTU output goes low and the ports enter high-impedance state.

(2) Select the reset-synchronized PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(3) Set reset-synchronized PWM mode.

(4) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(5) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(6) Start count operation by setting TSTRA (TSTRB).

(7) The reset-synchronized PWM waveform is output on compare match occurrence.

(8) An error occurs.

(9) Allow non-active level output by setting the pins as general output ports using the port mode registers (PMn), port registers (Pn), and port mode control registers (PMCn) of the I/O ports.

(10) Stop count operation by setting TSTRA (TSTRB). (MTU output becomes the initial reset-synchronized PWM output value.)

(11) Set normal mode (positive-phase MTU output goes low, and negative-phase output goes high).

(12) Initialize the pins with TIOR.

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(27) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in PWM Mode 1

Figure 16.180 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in PWM mode 1 after re-setting.

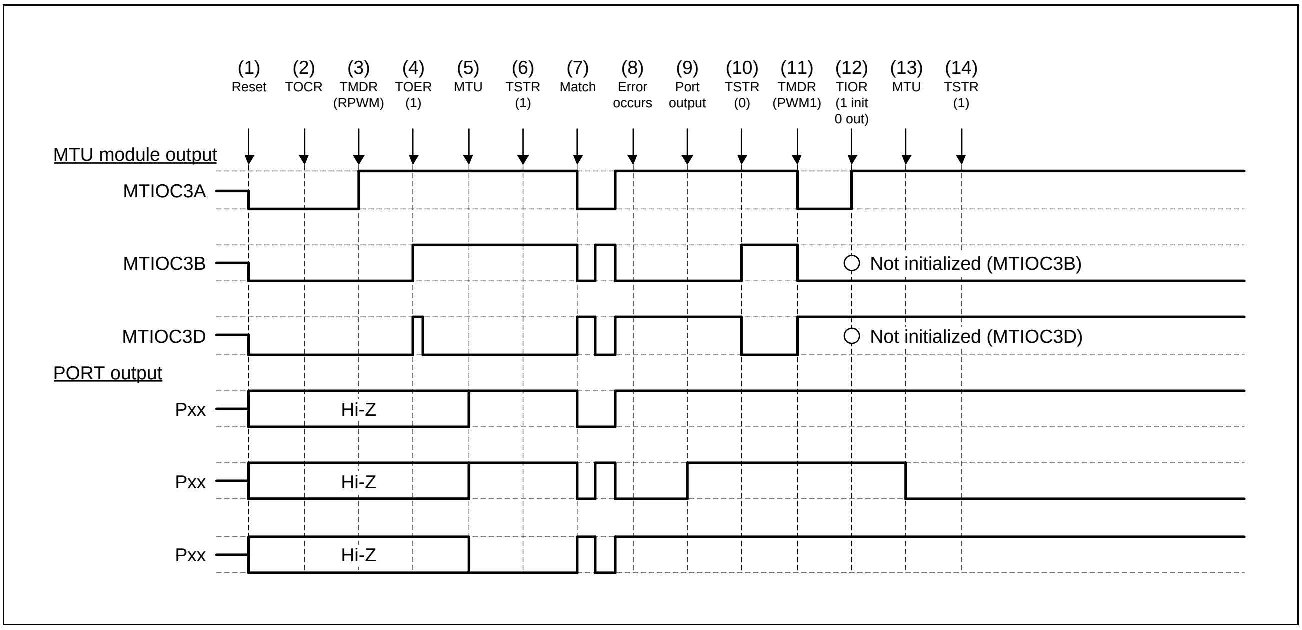


Figure 16.180 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in **Figure 16.179**.

(11) Set PWM mode 1 (positive-phase MTU output goes low, and negative-phase output goes high).

(12) Initialize the pins with TIOR. (In PWM mode 1, waveforms are not output to the MTIOCnB (MTIOCnD) pins. To output a specified level, make necessary settings for general output ports in the port mode registers (PMn) and port registers (Pn) of the I/O ports.)

(13) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(14) Restart operation by setting TSTRA (TSTRB).

(28) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Complementary PWM Mode

Figure 16.181 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in complementary PWM mode after re-setting.

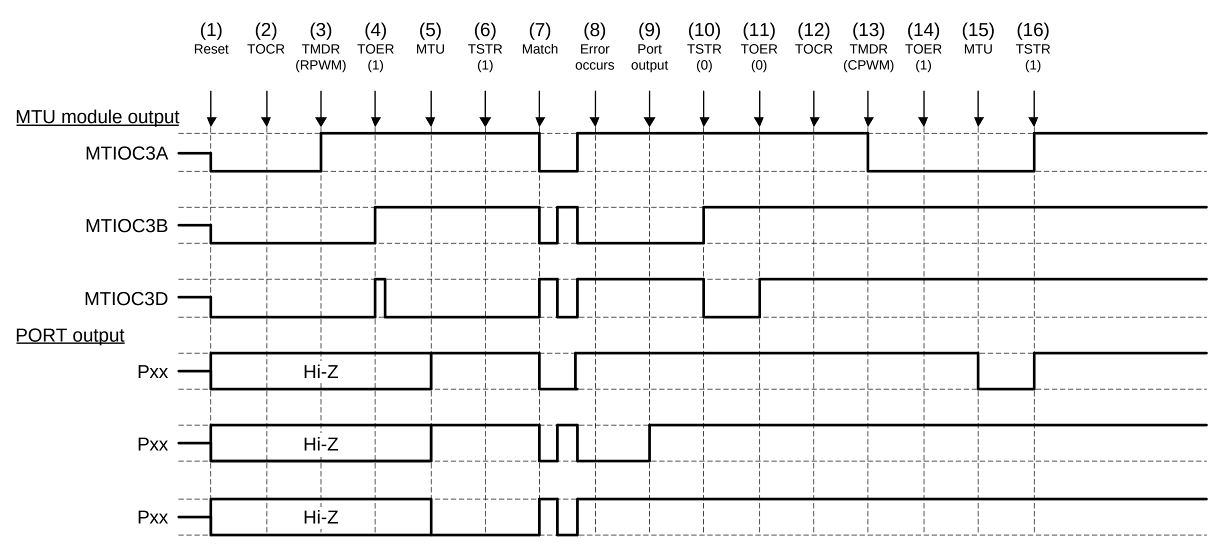


Figure 16.181 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in **Figure 16.179**.

(11) Disable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(12) Select the complementary PWM output level and enable or disable cyclic output with TOCR1A and TOCR2A (TOCR1B and TOCR2B).

(13) Set complementary PWM mode (MTU cyclic output pin goes low).

(14) Enable output in MTU3 and MTU4 (MTU6 and MTU7) with TOERA (TOERB).

(15) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(16) Restart operation by setting TSTRA (TSTRB).

(29) Operation When Error Occurs in Reset-Synchronized PWM Mode and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.182 shows a case in which an error occurs in reset-synchronized PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

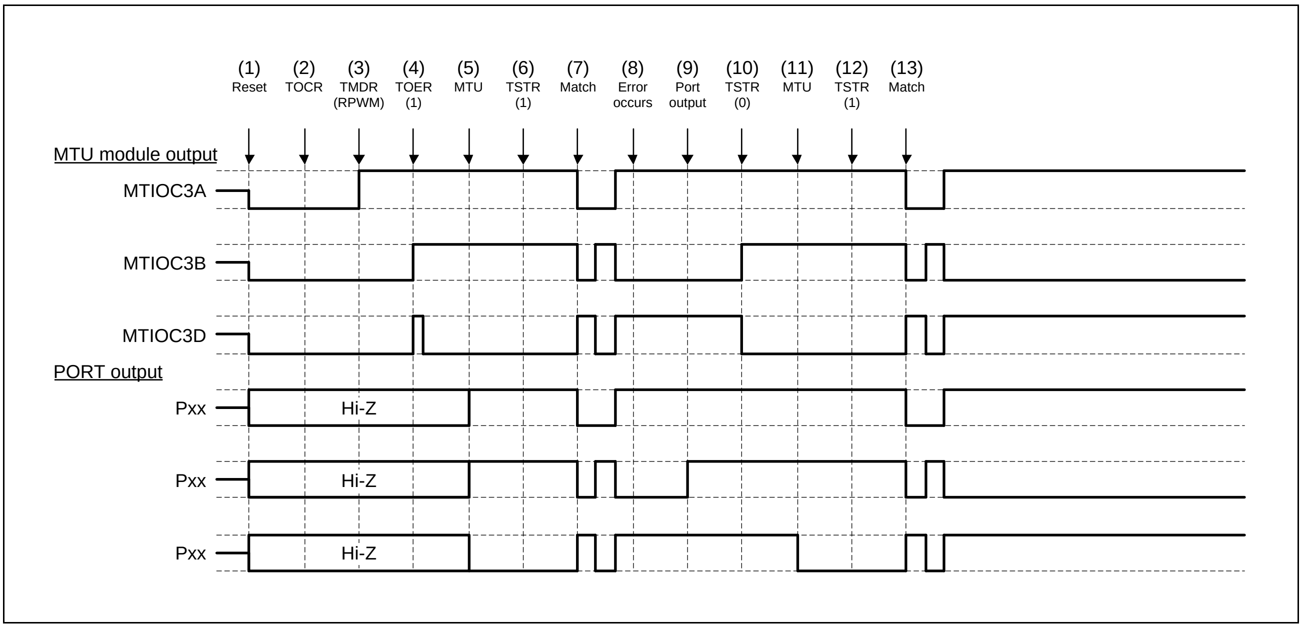


Figure 16.182 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in **Figure 16.179**.

(11) Set MTU output using the port mode control registers (PMCn) and Port function control registers (PFCm) corresponding to the GPIO ports.

(12) Restart operation by setting TSTRA (TSTRB).

(13) The reset-synchronized PWM waveform is output on compare match occurrence.

17. Port Output Enable 3 (POE3)

The port output enable 3 (POE3) can be used to place output pins for the MTU3a in the high-impedance state in response to various conditions.

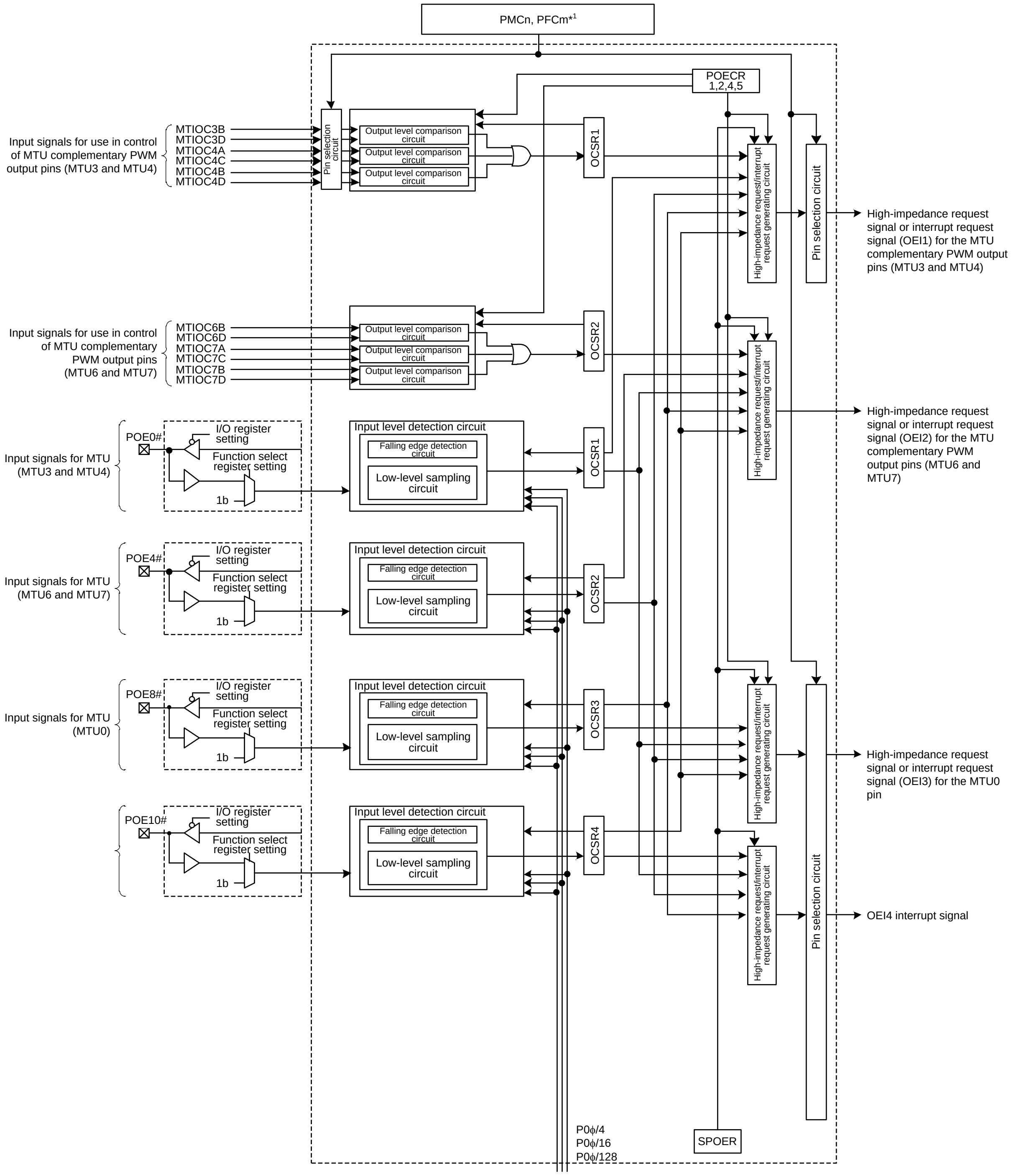
17.1 Overview

Table 17.1 lists the specifications of the POE3, and **Figure 17.1** shows a block diagram of the POE3.

Table 17.1 POE3 Specifications

<table>
<tr><th>Item</th><th>Description</th></tr>
<tr><td>Target pins to be placed in the high-impedance state</td><td>● MTU3a output pins
MTU0 pin (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D)
MTU3 pin (MTIOC3B, MTIOC3D)
MTU4 pin (MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D)
MTU6 pin (MTIOC6B, MTIOC6D)
MTU7 pin (MTIOC7A, MTIOC7B, MTIOC7C, MTIOC7D)</td></tr>
<tr><td>Conditions for the high-impedance state</td><td>● Setting pins as inputs: setting the POE0#, POE4#, POE8#, or POE10# pins as inputs (falling edge or low- level sampling).
● Short-circuits between output pins: A match (short circuit) between the output signal levels at the active level over one or more cycle on (the following combination of pins)
<table>
<tr><th></th><th>MTU complementary PWM output pins</th></tr>
<tr><td>1</td><td>MTIOC3B and MTIOC3D</td></tr>
<tr><td>2</td><td>MTIOC4A and MTIOC4C</td></tr>
<tr><td>3</td><td>MTIOC4B and MTIOC4D</td></tr>
<tr><td>4</td><td>MTIOC6B and MTIOC6D</td></tr>
<tr><td>5</td><td>MTIOC7A and MTIOC7C</td></tr>
<tr><td>6</td><td>MTIOC7B and MTIOC7D</td></tr>
</table>
● SPOER register setting being made</td></tr>
<tr><td>Function</td><td>● Each of the POE0#, POE4#, POE8#, and POE10# input pins can be set for falling edge, P0ϕ/4 × 16, P0ϕ/ 16 × 16, or P0ϕ/128 × 16 low-level sampling.
● Pins for the MTU complementary PWM output, and MTU0 pin can be placed in high-impedance state by POE0#, POE4#, POE8#, and POE10# pin falling-edge or low-level sampling.
● Pins for the MTU complementary PWM output can be placed in high-impedance state when output levels of the MTU complementary PWM output pins are compared and simultaneous active-level output continues for one cycle or more.
● Pins for the MTU complementary PWM output, and MTU0 can be placed in the high-impedance state by modifying the settings of the SPOER register of POE3.
● Interrupts can be generated by input-level sampling or output level comparison results.</td></tr>
</table>

The POE3 has input level detection circuits, pin selection circuits, output level comparison circuits, and a high-impedance request/interrupt request generating circuit as shown in **Figure 17.1**.



Note 1. Refer to chapter of GPIO.

Figure 17.1 POE3 Block Diagram

Table 17.2 shows input/output pins to be used by the POE3.

Table 17.2 POE3 Input/Output Pins

Pin Name	I/O	Description
POE0#	Input	Input pin for the request signal to place the MTU3 and MTU4 pins for MTU complementary PWM output in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU0, MTU6, and MTU7 pins in the high-impedance state.
POE4#	Input	Input pin for the request signal to place the MTU6 and MTU7 pins for MTU complementary PWM output in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU0, MTU3, and MTU4 pins in the high-impedance state.
POE8#	Input	Input pin for the request signal to place the pins for MTU0 in high-impedance state. In accord with register settings, this pin is also capable of placing the MTU3 and MTU4 pins or MTU6 and MTU7 pins for MTU complementary PWM output in high-impedance state.
POE10#	Input	In accord with register settings, this pin is also capable of placing the MTU3 and MTU4 pins or MTU6 and MTU7 pins for MTU complementary PWM output, and MTU0 pins in high-impedance state.

Table 17.3 shows output level comparisons with pin combinations.

Table 17.3 Pin Combinations

Pin Combination	I/O	Description
MTIOC3B and MTIOC3D	Output	The MTU3 and MTU4 pins for MTU complementary PWM output set in the M3SELR, M4SELR1, and M4SELR2 registers are placed in high-impedance state when both pins of a pair simultaneously output the active level*[1] for one or more cycles.of the POE3_CLKM_POE (P0ϕ). Pin combinations for output comparison and high-impedance control can be selected by registers of POE3. *Note 1.* The low level is output when the OLSP bit in TOCR1A of MTUn is 0 with the TOCS bit in TOCR1A of MTUn cleared to 0, or the high level is output when the OLSP bit is 1. Otherwise, the low level is output when the OLS3N, OLS3P, OLS2N, OLS2P, OLS1N, and OLS1P bits in TOCR2A of MTUn are 0 with the TOCS bit in TOCR1A of MTUn set to 1, or the high level is output when these bits are 1.
MTIOC4A and MTIOC4C	Output	
MTIOC4B and MTIOC4D	Output	
MTIOC6B and MTIOC6D	Output	The MTU6 and MTU7 pins for MTU complementary PWM output are placed in high-impedance state when both pins of a pair simultaneously output the active level*[1] for one or more cycles.of the POE3_CLKM_POE (P0ϕ). Pin combinations for output comparison and high-impedance control can be selected by registers of POE3. *Note 1.* The low level is output when the OLSP bit in TOCR1B of MTUn is 0 with the TOCS bit in TOCR1B of MTUn cleared to 0, or the high level is output when the OLSP bit is 1. Otherwise, the low level is output when the OLS3N, OLS3P, OLS2N, OLS2P, OLS1N, and OLS1P bits in TOCR2B of MTUn are 0 with the TOCS bit in TOCR1B of MTUn set to 1, or the high level is output when these bits are 1.
MTIOC7A and MTIOC7C	Output	
MTIOC7B and MTIOC7D	Output	

17.2 Register Descriptions

Table 17.4 shows the register configuration. The POE3 registers are initialized by a reset.

Base Address : H’0_1004_9800 (Cortex-A55 Address Space)
Base Address : H’4004_9800 (Cortex-M33 Address Space Non-Secure)
Base Address : H’5004_9800 (Cortex-M33 Address Space Secure)

Table 17.4 Register configuration

Register Name	Abbreviation	Offset Address	Access size
Input level control/status register 1	ICSR1	H’00	16
Output level control/status register 1	OCSR1	H’02	16
Input level control/status register 2	ICSR2	H’04	16
Output level control/status register 2	OCSR2	H’06	16
Input level control/status register 3	ICSR3	H’08	16
Software port output enable register	SPOER	H’0A	8
Port output enable control register 1	POECR1	H’0B	8
Port output enable control register 2	POECR2	H’0C	16
Port output enable control register 4	POECR4	H’10	16
Port output enable control register 5	POECR5	H’12	16
Input level control/status register 4	ICSR4	H’16	16

17.2.1 Input Level Control/Status Register 1 (ICSR1)

ICSR1 selects the input modes for the POE0# pins, controls the enable/disable of interrupts, and indicates status.

Address(es): H'1004_9800

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE0F	—	—	—	PIE1	—	—	—	—	—	—	POE0M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE0M[1:0]	All 0	R/W*1	POE0 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE0# input. 0 1: Accepts a request when POE0# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE0# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE0# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE1	0	R/W	Port Interrupt Enable 1 0: Interrupt requests disabled 1: Interrupt requests enabled
b11 to b9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE0F	0	R/(W)*2	POE0 Flag 0: Indicates that a high-impedance request has not been input to the POE0# pin. 1: Indicates that a high-impedance request has been input to the POE0# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE0M[1:0] Bits (POE0 Mode Select)

These bits select the input mode of the POE0# pin.

PIE1 Bit (Port Interrupt Enable 1)

This bit enables or disables interrupt requests when any one of the POE0F bit of the ICSR1 is set to 1.

POE0F Flag (POE0 Flag)

This flag indicates that a high-impedance request has been input to the POE0# pin.

[Setting condition]

- When the input set by POE0M[1:0] occurs at the POE0# pin

[Clearing condition]

- By writing 0 to POE0F after reading POE0F = 1

17.2.2 Input Level Control/Status Register 2 (ICSR2)

ICSR2 selects the input mode for the POE4# pin, controls the enable/disable of interrupts, and indicates status.

Address(es): H'1004_9804

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE4F	—	—	—	PIE2	—	—	—	—	—	—	POE4M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE4M[1:0]	All 0	R/W*1	POE4 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE4# input 0 1: Accepts a request when POE4# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE4# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE4# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE2	0	R/W	Port Interrupt Enable 2 0: Interrupt requests disabled 1: Interrupt requests enabled
b11 to b9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE4F	0	R/(W)*2	POE4 Flag 0: Indicates that a high-impedance request has not been input to the POE4# pin. 1: Indicates that a high-impedance request has been input to the POE4# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE4M[1:0] Bits (POE4 Mode Select)

These bits select the input mode of the POE4# pin.

PIE2 Bit (Port Interrupt Enable 2)

This bit enables or disables interrupt requests when the POE4F flag in ICSR2 is set to 1.

POE4F Flag (POE4 Flag)

This flag indicates that a high-impedance request has been input to the POE4# pin.

[Setting condition]

- When the input set by POE4M[1:0] occurs at the POE4# pin

[Clearing condition]

- By writing 0 to POE4F after reading POE4F = 1

17.2.3 Input Level Control/Status Register 3 (ICSR3)

ICSR3 selects the input mode for the POE8# pin, controls the enable/disable of interrupts, and indicates status.

Address(es): H’1004_9808

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE8F	—	—	POE8E	PIE3	—	—	—	—	—	—	POE8M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE8M[1:0]	All 0	R/W*1	POE8 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE8# input 0 1: Accepts a request when POE8# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE8# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE8# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE3	0	R/W	Port Interrupt Enable 3 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	POE8E	0	R/W*1	POE8 High- Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b11, b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE8F	0	R/(W)*2	POE8 Flag 0: Indicates that a high-impedance request has not been input to the POE8# pin. 1: Indicates that a high-impedance request has been input to the POE8# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE8M[1:0] Bits (POE8 Mode Select)

These bits select the input mode of the POE8# pin.

PIE3 Bit (Port Interrupt Enable 3)

This bit enables or disables interrupt requests when the POE8F bit in ICSR3 is set to 1.

POE8E Bit (POE8 High-Impedance Enable)

This bit specifies whether to place the corresponding pin in high-impedance state when the POE8F bit is set to 1.

POE8F Flag (POE8 Flag)

This flag indicates that a high-impedance request has been input to the POE8# pin.

[Setting condition]

- When the input set by POE8M[1:0] occurs at the POE8# pin

[Clearing condition]

- By writing 0 to POE8F after reading POE8F = 1

17.2.4 Input Level Control/Status Register 4 (ICSR4)

ICSR4 selects the POE10# pin input mode, controls the enable/disable of interrupts, and indicates status.

Address(es): H'1004_9816

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	POE10F	—	—	POE10E	PIE4	—	—	—	—	—	—	POE10M[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/(W)*2	R/W	R/W	R/W*1	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	POE10M[1:0]	All 0	R/W*1	POE10 Mode Select b1 b0 0 0: Accepts a request on the falling edge of POE10# input 0 1: Accepts a request when POE10# input has been sampled 16 times at P0ϕ/4 clock pulses and all are low level. 1 0: Accepts a request when POE10# input has been sampled 16 times at P0ϕ/16 clock pulses and all are low level. 1 1: Accepts a request when POE10# input has been sampled 16 times at P0ϕ/128 clock pulses and all are low level.
b7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	PIE4	0	R/W	Port Interrupt Enable 4 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	POE10E	0	R/W*1	POE10 High- Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b11, b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b12	POE10F	0	R/(W)*2	POE10 Flag 0: Indicates that a high-impedance request has not been input to the POE10# pin. 1: Indicates that a high-impedance request has been input to the POE10# pin.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bits are modified once, writing these bits does not modify their values until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

POE10M[1:0] Bits (POE10 Mode Select)

These bits select the input mode of the POE10# pin.

PIE4 Bit (Port Interrupt Enable 4)

This bit enables or disables interrupt requests when the POE10F bit is set to 1.

POE10E Bit (POE10 High-Impedance Enable)

This bit specifies whether to place the corresponding pin in high-impedance state when the POE10F bit is set to 1.

POE10F Bit (POE10 Flag)

This flag indicates that a request for the high-impedance state has been input to the POE10# pin.

[Setting condition]

- When the input set by POE10M[1:0] occurs at the POE10# pin

[Clearing condition]

- By writing 0 to POE10F after reading POE10F = 1

17.2.5 Output Level Control/Status Register 1 (OCSR1)

OCSR1 controls the enable/disable of output level comparison and interrupts, and indicates status.

Address(es): H'1004_9802

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF1	—	—	—	—	—	OCE1	OIE1	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b7 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	OIE1	0	R/W	Output Short Interrupt Enable 1 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	OCE1	0	R/W*1	Output Short High-Impedance Enable 1 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b14 to b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	OSF1	0	R/(W)*2	Output Short Flag 1 0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

OIE1 Bit (Output Short Interrupt Enable 1)

This bit enables or disables interrupt requests when the OSF1 bit in OCSR1 is set to 1.

OCE1 Bit (Output Short High-Impedance Enable 1)

This bit specifies whether to place the pins in high-impedance state when the OSF1 bit in OCSR1 is set to 1.

OSF1 Flag (Output Short Flag 1)

This flag indicates that any one of the three pairs of two-phase MTU3 and MTU4 pins for MTU complementary PWM output to be compared has simultaneously become an active level.

[Setting condition]

- When any one of the three pairs of two-phase outputs has simultaneously become an active level

[Clearing condition]

- By writing 0 to OSF1 after reading OSF1 = 1

17.2.6 Output Level Control/Status Register 2 (OCSR2)

OCSR2 controls the enable/disable of output level comparison and interrupts, and indicates status.

Address(es): H'1004_9806

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OSF2	—	—	—	—	—	OCE2	OIE2	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/(W)*2	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b7 to b0	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	OIE2	0	R/W	Output Short Interrupt Enable 2 0: Interrupt requests disabled 1: Interrupt requests enabled
b9	OCE2	0	R/W*1	Output Short High-Impedance Enable 2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b14 to b10	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	OSF2	0	R/(W)*2	Output Short Flag 2 0: Indicates that outputs have not simultaneously become an active level. 1: Indicates that outputs have simultaneously become an active level.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

Note 2. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

OIE2 Bit (Output Short Interrupt Enable 2)

This bit enables or disables interrupt requests when the OSF2 bit in OCSR2 is set to 1.

OCE2 Bit (Output Short High-Impedance Enable 2)

This bit specifies whether to place the pins in high-impedance state when the OSF2 bit in OCSR2 is set to 1.

OSF2 Flag (Output Short Flag 2)

This flag indicates that any one of the three pairs of two-phase MTU6 and MTU7 pins for MTU complementary PWM output to be compared has simultaneously become an active level.

[Setting condition]

- When any one of the three pairs of two-phase outputs has simultaneously become an active level

[Clearing condition]

- By writing 0 to OSF2 after reading OSF2 = 1

17.2.7 Software Port Output Enable Register (SPOER)

SPOER controls high-impedance state of the pins.

Address(es): H'1004_980A

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MTUCH0HIZ	MTUCH67HIZ	MTUCH34HIZ
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	MTUCH34HIZ	0	R/W	MTU3 or MTU4 Output High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b1	MTUCH67HIZ	0	R/W	MTU6 and MTU7 Output High- Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b2	MTUCH0HIZ	0	R/W	MTU0 Output High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

MTUCH34HIZ Bit (MTU3 or MTU4 Output High-Impedance Enable)

This bit specifies whether to place the MTU complementary PWM output pins (MTIOC3B, MTIOC3D, MTIOC4A, MTIOC4B, MTIOC4C, MTIOC4D) in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH34HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH34HIZ after reading MTUCH34HIZ = 1*[1]

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

MTUCH67HIZ Bit (MTU6 and MTU7 Output High-Impedance Enable)

This bit specifies whether to place the MTU complementary PWM output pins (MTIOC6B, MTIOC6D, MTIOC7A, MTIOC7B, MTIOC7C, and MTIOC7D) in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH67HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH67HIZ after reading MTUCH67HIZ = 1*[1]

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

MTUCH0HIZ Bit (MTU0 Output High-Impedance Enable)

This bit specifies whether to place the MTU0 pins in high-impedance state.

[Setting condition]

- By writing 1 to MTUCH0HIZ

[Clearing conditions]

- Reset
- By writing 0 to MTUCH0HIZ after reading MTUCH0HIZ = 1*[1]

Note 1. To write 0 to this bit, be sure to read 1 and then write 0.

17.2.8 Port Output Enable Control Register 1 (POECR1)

POECR1 controls high-impedance state of the MTU0 pins.

Address(es): H'1004_980B

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MTU0DZE	MTU0CZE	MTU0BZE	MTU0AZE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	MTU0AZE	0	R/W*1	MTIOC0A High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b1	MTU0BZE	0	R/W*1	MTIOC0B High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b2	MTU0CZE	0	R/W*1	MTIOC0C High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b3	MTU0DZE	0	R/W*1	MTIOC0D High-Impedance Enable 0: Does not place the pin in high-impedance state. 1: Places the pin in high-impedance state.
b7 to b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

MTU0AZE Bit (MTIOC0A High-Impedance Enable)

This bit specifies whether to place the MTIOC0A output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0BZE Bit (MTIOC0B High-Impedance Enable)

This bit specifies whether to place the MTIOC0B output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0CZE Bit (MTIOC0C High-Impedance Enable)

This bit specifies whether to place the MTIOC0C output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

MTU0DZE Bit (MTIOC0D High-Impedance Enable)

This bit specifies whether to place the MTIOC0D output for the MTU0 pin in high-impedance state when any of the POE8F flag in ICSR3, MTUCH0HIZ bit in SPOER or, as additionally specified in POECR5, the POEmF (n = 1, 2, 4; m = 0, 4, 10) flag in ICSRn, is set to 1.

17.2.9 Port Output Enable Control Register 2 (POECR2)

POECR2 controls high-impedance state of the MTU complementary PWM output pins (MTU3, MTU4, MTU6, and MTU7 pins).

Address(es): H'1004_980C

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MTU3B DZE	MTU4A CZE	MTU4B DZE	—	—	—	—	—	MTU6B DZE	MTU7A CZE	MTU7B DZE
Initial Value	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	MTU7BDZE	1	R/W*1	MTIOC7B/7D High-Impedance Enable*2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b1	MTU7ACZE	1	R/W*1	MTIOC7A/7C High-Impedance Enable*2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b2	MTU6BDZE	1	R/W*1	MTIOC6B/6D High-Impedance Enable*2 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	MTU4BDZE	1	R/W*1	MTIOC4B/4D High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b9	MTU4ACZE	1	R/W*1	MTIOC4A/4C High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b10	MTU3BDZE	1	R/W*1	MTIOC3B/3D High-Impedance Enable 0: Does not place the pins in high-impedance state. 1: Places the pins in high-impedance state.
b15 to b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

Note 2. Set this bit to 0 when MTU6 or MTU7 is not used.

MTU7BDZE Bit (MTIOC7B/7D High-Impedance Enable)

This bit specifies whether to place the MTIOC7B output and MTIOC7D output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU7ACZE Bit (MTIOC7A/7C High-Impedance Enable)

This bit specifies whether to place the MTIOC7A output and MTIOC7C output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU6BDZE Bit (MTIOC6B/6D High-Impedance Enable)

This bit specifies whether to place the MTIOC6B output and MTIOC6D output for the MTU3 pin in high-impedance state when any one of the OSF2 flag in OCSR2, POE4F flag in ICSR2, MTUCH67HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 1, 3, 4; m = 0, 8, 10) flag in ICSRn, is set to 1.

MTU4BDZE Bit (MTIOC4B/4D High-Impedance Enable)

This bit specifies whether to place the MTIOC4B output and MTIOC4D output for the MTU4 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

MTU4ACZE Bit (MTIOC4A/4C High-Impedance Enable)

This bit specifies whether to place the MTIOC4A output and MTIOC4C output for the MTU4 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

MTU3BDZE Bit (MTIOC3B/3D High-Impedance Enable)

This bit specifies whether to place the MTIOC3B output and MTIOC3D output for the MTU3 pin in high-impedance state when any one of the OSF1 flag in OCSR1, POE0F flag in ICSR1, MTUCH34HIZ bit in SPOER, OSTSTF flag in ICSR6 (when the OSTSTE bit is 1), or, as additionally specified in POECR4, the POEmF (n = 2 to 4; m = 4, 8, 10) flag in ICSRn, is set to 1.

17.2.10 Port Output Enable Control Register 4 (POECR4)

The POECR4 is used to extend the control conditions of the high-impedance state for the MTU3, MTU4, MTU6, and MTU7 pins for the MTU complementary PWM output.

For details about the targets and conditions of high-impedance control, see **Figure 17.2**.

Address(es): H'1004_9810

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IC4ADDMT67	IC3ADDMT67	—	IC1ADDMT67	—	—	—	—	IC4ADDMT34	IC3ADDMT34	IC2ADDMT34	—	—
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W	R/W*1	R/W	R/W	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b1	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b2	IC2ADDMT34ZE	0	R/W*1	MTU3 and MTU4 High- Impedance POE4F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b3	IC3ADDMT34ZE	0	R/W*1	MTU3 and MTU4 High- Impedance POE8F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b4	IC4ADDMT34ZE	0	R/W*1	MTU3 and MTU4 High- Impedance POE10F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b8 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0
b9	IC1ADDMT67ZE	0	R/W*1	MTU6 and MTU7 High- Impedance POE0F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b10	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b11	IC3ADDMT67ZE	0	R/W*1	MTU6 and MTU7 High- Impedance POE8F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b12	IC4ADDMT67ZE	0	R/W*1	MTU6 and MTU7 High- Impedance POE10F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b15 to b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

IC2ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE4F Add)

Adds the POE4F flag in ICSR2 (POE4#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC3ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE8F Add)

Adds the POE8F flag in ICSR3 (POE8#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC4ADDMT34ZE Bit (MTU3 and MTU4 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU3 and MTU4 pins (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4C/MTIOC4B/MTIOC4D).

IC1ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE0F Add)

Adds the POE0F flag in ICSR1 (POE0#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

IC3ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE8F Add)

Adds the POE8F flag in ICSR3 (POE8#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

IC4ADDMT67ZE Bit (MTU6 and MTU7 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU6, and MTU7 pins (MTIOC6B/MTIOC6D/MTIOC7A/MTIOC7C/MTIOC7B/MTIOC7D).

17.2.11 Port Output Enable Control Register 5 (POECR5)

The POECR5 is used to extend the control conditions of the high-impedance for the MTU0 pin.

For details about the targets and conditions of high-impedance control, see **Figure 17.2**.

Address(es): H'1004_9812

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	IC4ADDMT0ZE	—	IC2ADDMT0ZE	IC1ADDMT0ZE	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W	R/W*1	R/W*1	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b1	IC1ADDMT0ZE	0	R/W*1	MTU0 High-Impedance POE0F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b2	IC2ADDMT0ZE	0	R/W*1	MTU0 High-Impedance POE4F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b3	—	1	R/W	Reserved This bit is read as 1. The write value should be 1.
b4	IC4ADDMT0ZE	0	R/W*1	MTU0 High-Impedance POE10F Add 0: Does not add the pins to the high-impedance control conditions. 1: Adds the pins to the high-impedance control conditions.
b15 to b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Can be modified only once after a reset. After the bit is modified once, writing this bit does not modify its value until the register is reset.

IC1ADDMT0ZE Bit (MTU0 High-Impedance POE0F Add)

Adds the POE0F flag in ICSR1 (POE0#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC2ADDMT0ZE Bit (MTU0 High-Impedance POE4F Add)

Adds the POE4F flag in ICSR2 (POE4#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

IC4ADDMT0ZE Bit (MTU0 High-Impedance POE10F Add)

Adds the POE10F flag in ICSR4 (POE10#) to the high-impedance control conditions for the MTU0 pins (MTIOC0A, MTIOC0B, MTIOC0C, MTIOC0D).

17.3 Operation

Table 17.5 shows the target pins for high-impedance control and conditions to place the pins in high-impedance state.

Table 17.5 Target Pins and Conditions for High-Impedance Control (1/2)

Pins	Conditions	Detailed Conditions
MTU3 pins (MTIOC3B and MTIOC3D)	● Operation for detection of the POE0# input level ● Operation for comparison of the output levels on the MTIOC3B and MTIOC3D pins ● SPOER setting ● Additional conditions of the POECR4	MTU3BDZE• ((POE0F) +(OSF1•OCE1) +(MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU4 pins (MTIOC4A and MTIOC4C)	● Operation for detection of the POE0# input level ● Operation for comparison of the output levels on the MTIOC4A and MTIOC4C pins ● SPOER setting ● Additional conditions of the POECR4	MTU4ACZE• ((POE0F) +(OSF1•OCE1) + (MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU4 pins (MTIOC4B and MTIOC4D)	● Operation for detection of the POE0# input level ● Operation for comparison of the output levels on the MTIOC4B and MTIOC4D pins ● SPOER setting ● Additional conditions of the POECR4	MTU4BDZE• ((POE0F) +(OSF1•OCE1) +(MTUCH34HIZ) +(IC2ADDMT34ZE•POE4F) +(IC3ADDMT34ZE•POE8E•POE8F) +(IC4ADDMT34ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU6 pins (MTIOC6B and MTIOC6D)	● Operation for detection of the POE4# input level ● Operation for comparison of the output levels on the MTIOC6B and MTIOC6D pins ● SPOER setting ● Additional conditions of the POECR4	MTU6BDZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU7 pins (MTIOC7A and MTIOC7C)	● Operation for detection of the POE4# input level ● Operation for comparison of the output levels on the MTIOC7A and MTIOC7C pins ● SPOER setting ● Additional conditions of the POECR4	MTU7ACZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU7 pins (MTIOC7B and MTIOC7D)	● Operation for detection of the POE4# input level ● Operation for comparison of the output levels on the MTIOC7B and MTIOC7D pins ● SPOER setting ● Additional conditions of the POECR4	MTU7BDZE• ((POE4F) +(OSF2•OCE2) +(MTUCH67HIZ) +(IC1ADDMT67ZE•POE0F) +(IC3ADDMT67ZE•POE8E•POE8F) +(IC4ADDMT67ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0A)	● Operation for detection of the POE8# input level ● SPOER setting ● Additional conditions of the POECR5	MTU0AZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)

Table 17.5 Target Pins and Conditions for High-Impedance Control (2/2)

Pins	Conditions	Detailed Conditions
MTU0 pin (MTIOC0B)	● Operation for detection of the POE8# input level ● SPOER setting ● Additional conditions of the POECR5	MTU0BZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0C)	● Operation for detection of the POE8# input level ● SPOER setting ● Additional conditions of the POECR5	MTU0CZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)
MTU0 pin (MTIOC0D)	● Operation for detection of the POE8# input level ● SPOER setting ● Additional conditions of the POECR5	MTU0DZE• ((POE8F•POE8E) +(MTUCH0HIZ) +(IC1ADDMT0ZE•POE0F) +(IC2ADDMT0ZE•POE4F) +(IC4ADDMT0ZE•ICSR4.POE10E•ICSR4.POE10F)

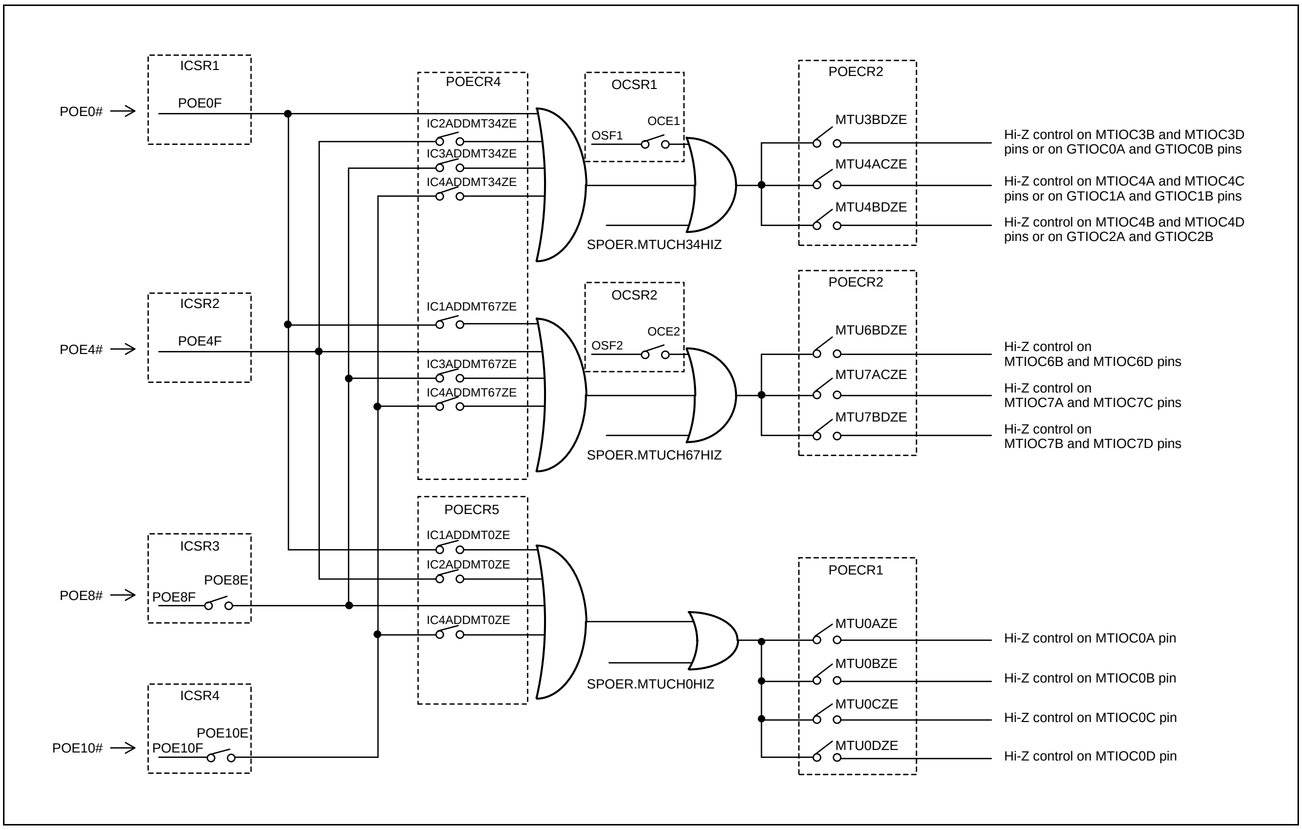


Figure 17.2 Target Pins and Conditions for High-Impedance Control

High-impedance requests to individual pins can be controlled by the settings in the POECR1 to POECR2 registers.

The following input pins can be added to high-impedance control conditions by the settings in the POECR4 to POECR5 registers: input pins other than the POE0# pin for the MTU3 and MTU4 pins; input pins other than the POE4# pin for the MTU6 and MTU7 pins; input pins other than the POE8# pin for the MTU0 pins.

For example, setting the IC2ADDMT34ZE bit in POECR4 to 1 outputs high-impedance requests to the MTU3 and MTU4 pins even when the POE4# input is detected.

High-impedance requests due to the detection of POE8# input or POE10# input can be controlled by the settings in the ICSR3 to ICSR4 registers. (The ICSR1 and ICSR2 registers do not control enabling or disabling of output of high-impedance requests.)

High-impedance requests to the MTU3, MTU4, MTU6, and MTU7 pins as the results of output level comparison can be controlled by the settings in the OCSR1 and OCSR2 registers.

17.3.1 MTU Pin Selection

In this LSI, each terminal function for MTU is assigned to multiple ports. **Table 17.6** shows the correspondence table of MTU terminal functions and compatible ports. The pin used as the MTU must be set separately in the general-purpose I / O port register. (Set by PMCn and PFCm registers of IO_TOP)

Table 17.6 Correspondence between MTU Pins (1/2)

MTU Pin Functions	Corresponding Ports
MTCLKA	P14_0
	P22_0
	P48_0
MTCLKB	P14_1
	P22_1
	P48_1
MTCLKC	P15_0
	P23_0
	P48_2
MTCLKD	P15_1
	P23_1
	P48_3
MTIOC0A	P0_0
	P34_0
MTIOC0B	P0_1
	P34_1
MTIOC0C	P1_0
	P35_0
MTIOC0D	P1_1
	P35_1
MTIOC1A	P2_0
	P6_0
	P19_0
MTIOC1B	P2_1
	P6_0
	P19_1
MTIOC2A	P3_0
	P9_0
	P18_0
MTIOC2B	P3_1
	P9_1
	P18_1
MTIOC3A	P32_0
	P44_0
MTIOC3B	P32_1
	P44_1
MTIOC3C	P36_0
	P44_2

Table 17.6 Correspondence between MTU Pins (2/2)

MTU Pin Functions	Corresponding Ports
MTIOC3D	P36_1
	P44_3
MTIOC4A	P38_0
MTIOC4B	P38_1
MTIOC4C	P39_0
MTIOC4D	P39_1
MTIC5U	P7_0
	P40_0
MTIC5V	P7_1
	P40_1
MTIC5W	P7_2
	P40_2
MTIOC6A	P10_0
MTIOC6B	P10_1
MTIOC6C	P11_0
MTIOC6D	P11_1
MTIOC7A	P4_0
	P42_0
MTIOC7B	P4_1
	P42_1
MTIOC7C	P5_0
	P42_2
MTIOC7D	P5_1
	P42_3
MTIOC8A	P26_0
	P43_0
MTIOC8B	P26_1
	P43_1
MTIOC8C	P27_0
	P43_2
MTIOC8D	P27_1
	P43_3

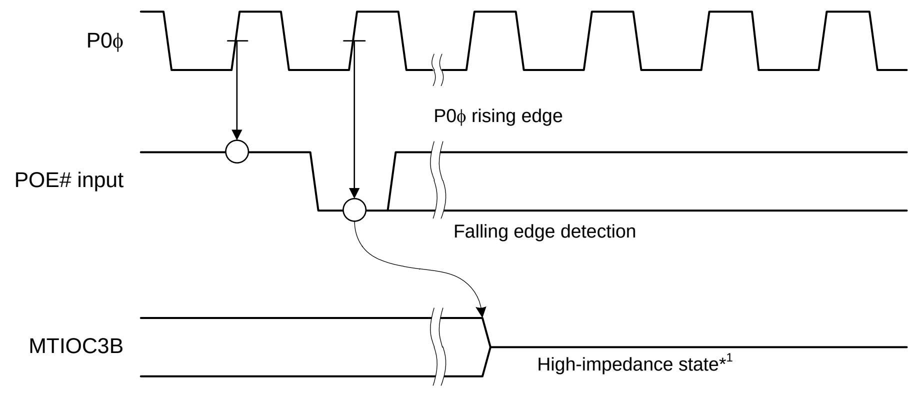
17.3.2 Input Level Detection Operation

If the input conditions set by ICSR1 to ICSR4 occur on the POE0#, POE4#, POE8#, and POE10# pins, the MTU3 and MTU4 or MTU6 and MTU7 pins for the MTU complementary PWM output, and MTU0 pin are placed in high-impedance state. Note however, that these pins are still placed in the high-impedance state even when the MTU functions are not selected for the pins.

(1) Falling Edge Detection

When a change from a high to low level is input to the POE0#, POE4#, POE8#, and POE10# pins, the pins for the MTU complementary PWM output, and pin functions multiplexed with the MTU0 pins are placed in highimpedance state.

Figure 17.3 shows a sample timing after the level changes in input to the POE0#, POE4#, POE8#, and POE10# pins until the respective pins enter high-impedance state.

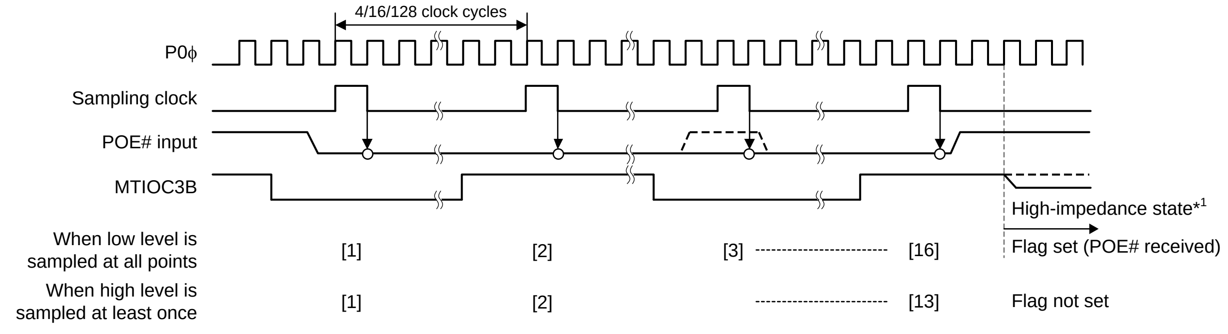


Note 1. Other pins also enter the high-impedance state at the same timing.

Figure 17.3 Falling Edge Detection

(2) Low-Level Detection

Figure 17.4 shows the low-level detection operation. Sixteen continuous low levels are sampled with the sampling clock selected by ICSR1 to ICSR4. If even one high level is detected during this interval, the low level is not accepted. The timing when pins for the MTU complementary PWM output, and MTU0 pins enter the highimpedance state after the sampling clock is input is the same in both falling-edge detection and in low-level detection.



Note 1. Other pins also enter the high-impedance state at the same timing.

Figure 17.4 Low-Level Detection Operation

17.3.3 Output Level Compare Operation

Figure 17.5 shows an example of the output level compare operation for the combination of MTIOC3B and MTIOC3D. The operation is the same for the other pin combinations.

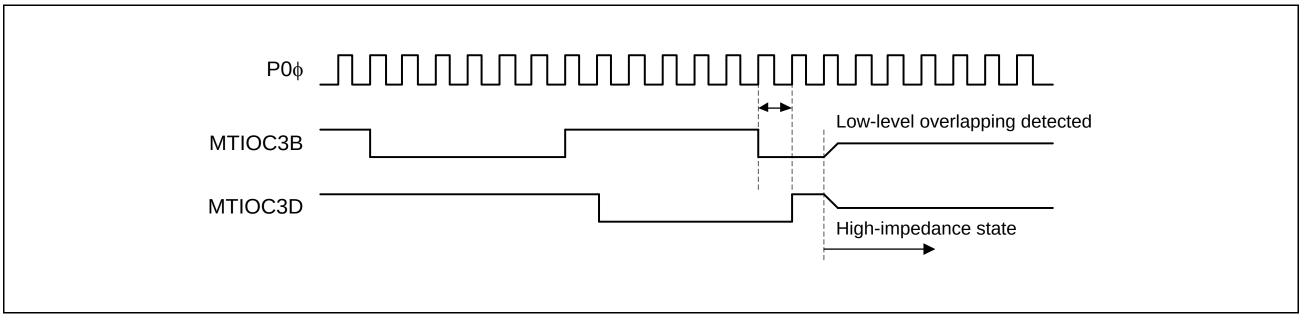


Figure 17.5 Output Level Compare Operation

17.3.4 High-Impedance Control Using Registers

The high-impedance state of the MTU pins (MTU0, MTU3, MTU4, MTU6, and MTU7) can be directly controlled by using the software port output enable register (SPOER).

For instance, setting the MTUCH34HIZ bit in SPOER to 1 places the MTU3 and MTU4 pins specified by the port output enable control register 2 (POECR2) in the high-impedance state.

The high-impedance state of other pins can also be controlled by setting the appropriate bits in SPOER.

17.3.5 Additional Functions for Controlling High-Impedance States

Settings in port enable registers 4 to 5 (POECR4 and POECR5) can add further high-impedance control conditions for the MTU complementary PWM output and MTU0 pins.

For instance, the settings listed below can be added as high-impedance control conditions for the MTU3 and MTU4 pins.

- Setting the IC2ADDMT34ZE bit in POECR4 to 1 adds the input level detection by the POE4#
- Setting the IC3ADDMT34ZE bit in POECR4 to 1 and adds the input level detection by the POE8#
- Setting the IC4ADDMT34ZE bit in POECR4 to 1 and adds the input level detection by the POE10# (ICSR4.POE10F)

The high-impedance state of other pins can also be controlled by setting the appropriate bits in the POECR4 and POECR5.

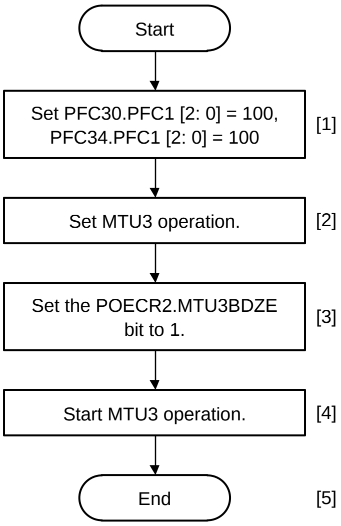
17.3.6 Release from High-Impedance State

MTU pins which have entered high-impedance state due to input-level detection can be released from the state either by returning them to their initial state with a reset, or by clearing all of the ICSR1.POE0F, ICSR2.POE4F, ICSR3.POE8F, and ICSR4.POE10F flags. However, note that when low-level sampling is selected with the ICSR1.POE0M[1:0], ICSR2.POE4M[1:0], ICSR3.POE8M[1:0], ICSR4.POE10M[1:0], and ICSR5.POE10M[1:0] bits, just writing 0 to a flag is ignored (the flag is not cleared); flags can be cleared by writing 0 to it only after a high level is input to the POE0#, POE4#, POE8#, and POE10# pins and is sampled.

MTU pins which have entered high-impedance state due to output level detection can be released from the state either by returning them to their initial state with a reset, or by clearing the OCSR1.OSF1 flag or the OCSR2.OSF2 flag.

17.4 POE3 Setting Procedure

Figure 17.6 shows the POE3 configuration procedure. As an example, high impedance control by comparing the output levels of the MTU 3 terminals (MTIOC3B / MTIOC3D) is shown. In **Figure 17.6**, select P32_1 for the MTIOC3B terminal and P36_1 for the MTIOC3D terminal.



Set the MTU3 pin (P32_1, P36_1) to the control mode (combined function) with PMCn (port mode control register) in advance, and then make the following settings.

[1] Select P32_1 as the MTIOC3B pin and P36_1 as the MTIOC3D pin by setting the PFC30 register and PFC34 register of the general-purpose I/O port (IO_TOP).

[2] Set MTU3 operation.

[3] Enable the high -impedance control for MTIOC3B and MTIOC3D pin by setting POECR2.MTU3BDZE bit to 1.

[4] Start MTU3 operation.

After making the above settings, if MTIOC3B (P32_1) and MTIOC3D (P36_1) go low at the same time, the same terminal becomes high impedance and the OCSR1.OSF1 bit is set.

For the method for releasing high-impedance, refer to **Section 17.3.6, Release from High-Impedance State**.

Figure 17.6 Procedure for Setting POE3

17.5 Interrupts

The POE3 issues a request to generate an interrupt when the specified condition is satisfied during input level detection or output level comparison. **Table 17.7** shows the interrupt sources and their conditions.

Table 17.7 Interrupt Sources and Conditions

Name	Interrupt Source	Interrupt Flag	Condition
OEI1	Output enable interrupt 1	POE0F and OSF1	PIE1•POE0F+OIE1•OSF1
OEI2	Output enable interrupt 2	POE4F and OSF2	PIE2•POE4F+OIE2•OSF2
OEI3	Output enable interrupt 3	POE8F	PIE3•POE8F
OEI4	Output enable interrupt 4	ICSR4.POE10F	PIE4•ICSR4.POE10F

17.6 Usage Notes

17.6.1 High-Impedance Control when MTU6 and MTU7 are not Used

When MTU6 and MTU7 are not to be used, set the POECR2.MTU6BDZE, MTU7ACZE, and MTU7BDZE bits to 0 to disable high-impedance control.

18. General PWM Timer (GPT)

18.1 Overview

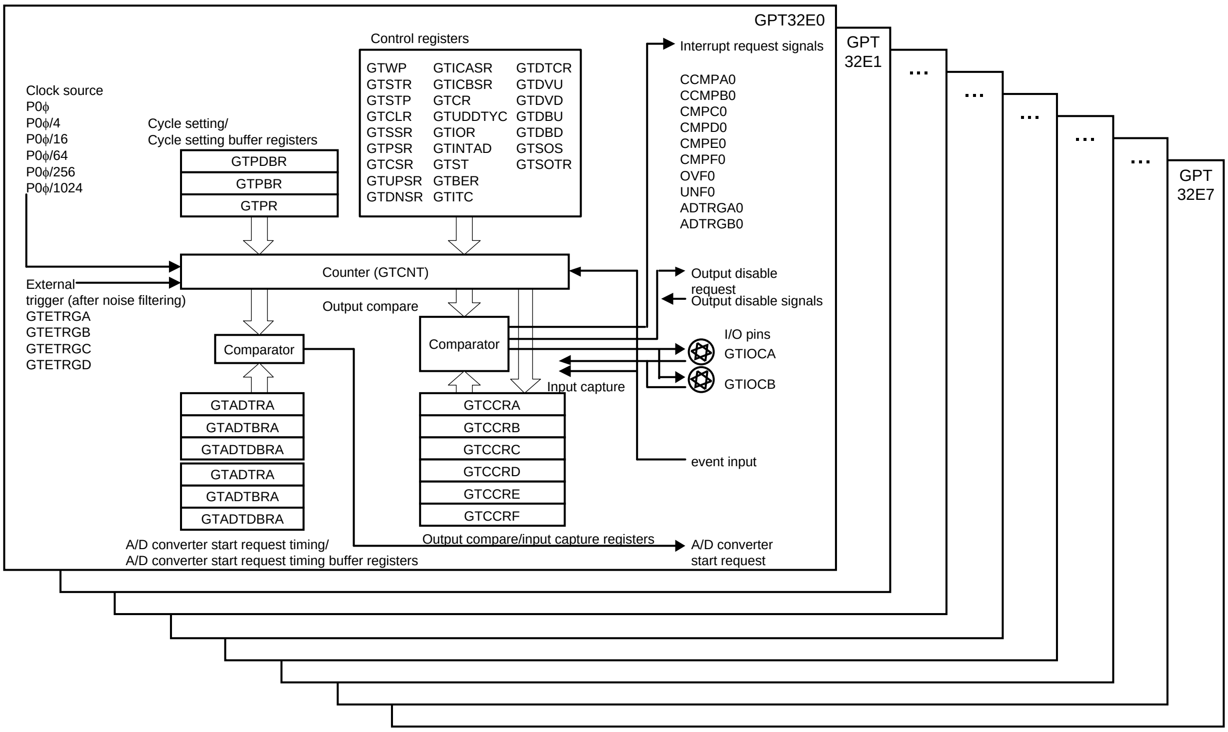
This LSI has a general purpose PWM timer (GPT) composed of 8 channels of 32-bit timer (GPT32E). **Table 18.1** lists the GPT specifications, **Table 18.2** shows the GPT functions, **Figure 18.1** shows the block diagram, and **Table 18.3** lists the I/O pins.

Table 18.1 GPT specifications

Parameter	Specifications
Functions	● 32 bits × 8 channels ● Up-counting or down-counting (saw waves) or up/down-counting (triangle waves) for each counter. ● Clock sources independently selectable for each channel ● Two I/O pins per channel ● Two output compare/input capture registers per channel ● For the two output compare/input capture registers of each channel, four registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use. ● In output compare operation, buffer switching can be at crests or troughs, enabling the generation of laterally asymmetric PWM waveforms. ● Registers for setting up frame cycles in each channel (with capability for generating interrupts at overflow or underflow) ● Generation of dead times in PWM operation ● Synchronous starting, stopping and clearing counters for arbitrary channels ● Starting, stopping, clearing and up/down counters in response to input level comparison ● Starting, clearing, stopping and up/down counters in response to a maximum of four external triggers ● Output pin disable function by dead time error and detected short-circuits between output pins ● A/D converter start triggers can be generated (GPT32E0 to GPT32E3) ● Enables the noise filter for input capture and external trigger operation

Table 18.2 GPT functions

Parameter		GPT32E
Count clock		P0ϕ P0ϕ/4 P0ϕ/16 P0ϕ/64 P0ϕ/256 P0ϕ/1024
Output compare/input capture registers (GTCCR)		GTCCRA GTCCRB
Compare/buffer registers		GTCCRC GTCCRD GTCCRE GTCCRF
Cycle setting register		GTPR
Cycle setting buffer registers		GTPBR GTPDBR
I/O pins		GTIOCA GTIOCB
External trigger input pin		GTETRGA GTETRGB GTETRGC GTETRGD
Counter clear sources		GTPR register compare match, input capture, input pin status, or input on the GTETRGA, GTETRGB, GTETRGC, or GTETRGD pins
Compare match output	Low output	Available
	High output	Available
	Toggle output	Available
Input capture function		Available
Automatic addition of dead time		Available
PWM mode		Available
Phase count function		Available
Buffer operation		Double buffer
One-shot operation		Available
DMA activation		All the interrupt sources
A/D converter start trigger		Compare match of GTADTRA or GTADTRB (Channel 0 to 3)
Interrupt sources		10 sources ● GTCCRA compare match/input capture (CCMPAn) ● GTCCRB compare match/input capture (CCMPBn) ● GTCCRC compare match (CMPCn) ● GTCCRD compare match (CMPDn) ● GTCCRE compare match (CMPEn) ● GTCCRF compare match (CMPFn) ● GTADTRA compare match (ADTRGAn) ● GTADTRB compare match (ADTRGBn) ● GTCNT overflow (GTPR compare match) (OVFn) ● GTCNT underflow (UNFn)
Interrupt skipping function		Skips GTCNT overflows (GTPR compare match) (OVFn)/GTCNT underflow (UNFn) interrupts (with interlocking function for other interrupts or A/D conversion requests).
Noise filtering function		Available



GTWP:	General PWM Timer Write-Protection Register
GTSTR:	General PWM Timer Software Start Register
GTSTP:	General PWM Timer Software Stop Register
GTCLR:	General PWM Timer Software Clear Register
GTSSR:	General PWM Timer Start Source Select Register
GTPSR:	General PWM Timer Stop Source Select Register
GTCSR:	General PWM Timer Clear Source Select Register
GTUPSR:	General PWM Timer Up Count Source Select Register
GTDNSR:	General PWM Timer Down Count Source Select Register
GTICASR:	General PWM Timer Input Capture Source Select Register A
GTICBSR:	General PWM Timer Input Capture Source Select Register B
GTCR:	General PWM Timer Control Register
GTUDDTYC:	General PWM Timer Count Direction and Duty Setting Register
GTIOR:	General PWM Timer I/O Control Register
GTINTAD:	General PWM Timer Interrupt Output Setting Register
GTST:	General PWM Timer Status Register
GTBER:	General PWM Timer Buffer Enable Register
GTITC:	General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register
GTCNT:	General PWM Timer Counter
GTCCRA:	General PWM Timer Compare Capture Register A
GTCCRB:	General PWM Timer Compare Capture Register B
GTCCRC:	General PWM Timer Compare Capture Register C
GTCCRD:	General PWM Timer Compare Capture Register D
GTCCRE:	General PWM Timer Compare Capture Register E
GTCCRF:	General PWM Timer Compare Capture Register F
GTPR:	General PWM Timer Cycle Setting Register
GTPBR:	General PWM Timer Cycle Setting Buffer Register
GTPDBR:	General PWM Timer Cycle Setting Double-Buffer Register
GTADTRA:	General PWM Timer A/D Converter Start Request Timing Register A
GTADTBRA:	General PWM Timer A/D Converter Start Request Timing Buffer Register A
GTADTDBRA:	General PWM Timer A/D Converter Start Request Timing Double-Buffer Register A
GTADTRB:	General PWM Timer A/D Converter Start Request Timing Register B
GTADTBRB:	General PWM Timer A/D Converter Start Request Timing Buffer Register B
GTADTDBRB:	General PWM Timer A/D Converter Start Request Timing Double-Buffer Register B
GTDTCR:	General PWM Timer Dead Time Control Register
GTDVU:	General PWM Timer Dead Time Value Register U
GTDVD:	General PWM Timer Dead Time Value Register D
GTDBU:	General PWM Timer Dead Time Buffer Register U
GTDBD:	General PWM Timer Dead Time Buffer Register D
GTSOS:	General PWM Timer Output Protection Function Status Register
GTSOTR:	General PWM Timer Output Protection Function Temporary Release Register

Figure 18.1 GPT block diagram

Table 18.3 GPT I/O pins

Channel	Pin name	I/O	Function
Shared	GTETRGA	Input	External trigger input pin A (after noise filtering)
	GTETRGB	Input	External trigger input pin B (after noise filtering)
	GTETRGC	Input	External trigger input pin C (after noise filtering)
	GTETRGD	Input	External trigger input pin D (after noise filtering)
GPT32E0	GTIOC0A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC0B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E1	GTIOC1A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC1B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E2	GTIOC2A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC2B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E3	GTIOC3A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC3B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E4	GTIOC4A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC4B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E5	GTIOC5A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC5B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E6	GTIOC6A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC6B	I/O	GTCCRB register input capture input/output compare output/PWM output pin
GPT32E7	GTIOC7A	I/O	GTCCRA register input capture input/output compare output/PWM output pin
	GTIOC7B	I/O	GTCCRB register input capture input/output compare output/PWM output pin

18.2 Register Descriptions

Table 18.4 lists the registers in the GPT.

Base Address: H’0_1004_8000 (Cortex-A55 Address Space)

Base Address: H’4004_8000 (Cortex-M33 Address Space Non-Secure)

Base Address: H’5004_8000 (Cortex-M33 Address Space Secure)

Table 18.4 GPT registers (1/2)

Module symbol	Register name	Register symbol	Reset value	Offset Address (m = 0 to 7)	Access size
GPT32Em (m=0 to 7)	General PWM Timer Write-Protection Register	GTWP	H’00000000	H’00+H’0100×m	32
	General PWM Timer Software Start Register	GTSTR	H’00000000	H’04+H’0100×m	32
	General PWM Timer Software Stop Register	GTSTP	H’FFFFFFFF	H’08+H’0100×m	32
	General PWM Timer Software Clear Register	GTCLR	H’00000000	H’0C+H’0100×m	32
	General PWM Timer Start Source Select Register	GTSSR	H’00000000	H’10+H’0100×m	32
	General PWM Timer Stop Source Select Register	GTPSR	H’00000000	H’14+H’0100×m	32
	General PWM Timer Clear Source Select Register	GTCSR	H’00000000	H’18+H’0100×m	32
	General PWM Timer Up Count Source Select Register	GTUPSR	H’00000000	H’1C+H’0100×m	32
	General PWM Timer Down Count Source Select Register	GTDNSR	H’00000000	H’20+H’0100×m	32
	General PWM Timer Input Capture Source Select Register A	GTICASR	H’00000000	H’24+H’0100×m	32
	General PWM Timer Input Capture Source Select Register B	GTICBSR	H’00000000	H’28+H’0100×m	32
	General PWM Timer Control Register	GTCR	H’00000000	H’2C+H’0100×m	32
	General PWM Timer Count Direction and Duty Setting Register	GTUDDTYC	H’00000001	H’30+H’0100×m	32
	General PWM Timer I/O Control Register	GTIOR	H’00000000	H’34+H’0100×m	32
	General PWM Timer Interrupt Output Setting Register	GTINTAD	H’00000000	H’38+H’0100×m	32
	General PWM Timer Status Register	GTST	H’00008000	H’3C+H’0100×m	32
	General PWM Timer Buffer Enable Register	GTBER	H’00000000	H’40+H’0100×m	32
	General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register	GTITC	H’00000000	H’44+H’0100×m	32
	General PWM Timer Counter	GTCNT	H’00000000	H’48+H’0100×m	32
	General PWM Timer Compare Capture Register A	GTCCRA	H’FFFFFFFF	H’4C+H’0100×m	32
	General PWM Timer Compare Capture Register B	GTCCRB	H’FFFFFFFF	H’50+H’0100×m	32
	General PWM Timer Compare Capture Register C	GTCCRC	H’FFFFFFFF	H’54+H’0100×m	32
	General PWM Timer Compare Capture Register E	GTCCRE	H’FFFFFFFF	H’58+H’0100×m	32
	General PWM Timer Compare Capture Register D	GTCCRD	H’FFFFFFFF	H’5C+H’0100×m	32
	General PWM Timer Compare Capture Register F	GTCCRF	H’FFFFFFFF	H’60+H’0100×m	32
	General PWM Timer Cycle Setting Register	GTPR	H’FFFFFFFF	H’64+H’0100×m	32
	General PWM Timer Cycle Setting Buffer Register	GTPBR	H’FFFFFFFF	H’68+H’0100×m	32
	General PWM Timer Cycle Setting Double-Buffer Register	GTPDBR	H’FFFFFFFF	H’6C+H’0100×m	32
	A/D Converter Start Request Timing Register A	GTADTRA	H’FFFFFFFF	H’70+H’0100×m	32

Table 18.4 GPT registers (2/2)

Module symbol	Register name	Register symbol	Reset value	Address (m = 0 to 7)	Access size
GPT32Em (m=0 to 7)	A/D Converter Start Request Timing Buffer Register A	GTADTBRA	H'FFFFFFFF	H'74+H'0100×m	32
	A/D Converter Start Request Timing Double-Buffer Register A	GTADTDBRA	H'FFFFFFFF	H'78+H'0100×m	32
	A/D Converter Start Request Timing Register B	GTADTRB	H'FFFFFFFF	H'7C+H'0100×m	32
	A/D Converter Start Request Timing Buffer Register B	GTADTBRB	H'FFFFFFFF	H'80+H'0100×m	32
	A/D Converter Start Request Timing Double-Buffer Register B	GTADTDBRB	H'FFFFFFFF	H'84+H'0100×m	32
	General PWM Timer Dead Time Control Register	GTDTCR	H'00000000	H'88+H'0100×m	32
	General PWM Timer Dead Time Value Register U	GTDVU	H'FFFFFFFF	H'8C+H'0100×m	32
	General PWM Timer Dead Time Value Register D	GTDVD	H'FFFFFFFF	H'90+H'0100×m	32
	General PWM Timer Dead Time Buffer Register U	GTDBU	H'FFFFFFFF	H'94+H'0100×m	32
	General PWM Timer Dead Time Buffer Register D	GTDBD	H'FFFFFFFF	H'98+H'0100×m	32
	General PWM Timer Output Protection Function Status Register	GTSOS	H'00000000	H'9C+H'0100×m	32
	General PWM Timer Output Protection Function Temporary Release Register	GTSOTR	H'00000000	H'A0+H'0100×m	32

18.2.1 General PWM Timer Write-Protection Register (GTWP)

Address(es): GPT32Em.GTWP H’1004_8000 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]								—	—	—	—	—	—	—	WP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	WP	0	R/W	Register Write Disable 0: Enable writes to the register 1: Disable writes to the register.
b7 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15 to b8	PRKEY[7:0]	All 0	R/W	GTWP Key Code When H’A5 is written to these bits, the writes to the WP bit are permitted. These bits are read as 0.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

To prevent accidental changes, the GTWP enables or disables writing to following registers:

- GTSSR
- GTPSR
- GTCSR
- GTUPSR
- GTDNSR
- GTICASR
- GTICBSR
- GTCR
- GTUDDTYC
- GTIOR
- GTINTAD
- GTST
- GTBER
- GTITC
- GTCNT
- GTCCRA
- GTCCRB
- GTCCRC
- GTCCRD
- GTCCRE
- GTCCRF
- GTPR
- GTPBR
- GTPDBR
- GTADTRA
- GTADTBRA
- GTADTDBRA
- GTADTRB
- GTADTBRB
- GTADTDBRB
- GTDTCR
- GTDVU
- GTDVD
- GTDBU
- GTDBD
- GTSOS
- GTSOTR.

18.2.2 General PWM Timer Software Start Register (GTSTR)

Address(es): GPT32Em.GTSTR H'1004_8004 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CSTRT7	CSTRT6	CSTRT5	CSTRT4	CSTRT3	CSTRT2	CSTRT1	CSTRT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The GTSTR starts the GTCNT counter operation for each channel x, where x = 0 to 7.

The GTSTR bit number represents the channel number. The GTSTR register is shared by all of the channels. The GTCNT counter starts for the channel associated with the GTSTR bit number where 1 is written. Writing 0 has no effect on the status of the GTCNT counter and the value of GTSTR register.

CSTRTx bit (channel x GTCNT Count Start) (x = 0 to 7)

The CSTRTx bit starts channel x of the GTCNT counter operation. Writing to GTSTR.CSTRTx bit has no effect unless GPTx.GTSSR.CSTRT bit is set to 1.

Read data shows the counter status of each channel (GTCR.CST bit). Zero means the counter stops and 1 means the counter is running.

18.2.3 General PWM Timer Software Stop Register (GTSTP)

Address(es): GPT32Em.GTSTP H’1004_8008 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CSTOP 7	CSTOP 6	CSTOP 5	CSTOP 4	CSTOP 3	CSTOP 2	CSTOP 1	CSTOP 0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The GTSTP stops the GTCNT counter operation for each channel x, where x = 0 to 7.

The GTSTP bit number represents the channel number. The GTSTP register is shared by all of the channels. The GTCNT counter stops for the channel associated with the GTSTP bit number where 1 is written. Writing 0 has no effect on the status of GTCNT counter and the value of GTSTP register.

CSTOPx bit (channel x GTCNT Count Stop) (x = 0 to 7)

The CSTOPx bit stops channel x of the GTCNT counter operation. Writing to GTSTP.CSTOPx bit has no effect unless GPTx.GTPSR.CSTOP bit is set to 1. Read data shows the counter status of each channel (invert of GTCR.CST bit). Zero means the counter is running and 1 means the counter stops.

18.2.4 General PWM Timer Software Clear Register (GTCLR)

Address(es): GPT32Em.GTCLR H’1004_800C + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CCLR7	CCLR6	CCLR5	CCLR4	CCLR3	CCLR2	CCLR1	CCLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTCLR is a write-only register that clears the GTCNT counter operation for each channel x, where x = 0 to 7. The GTCLR bit number represents the channel number. The GTCLR register is shared by all of the channels. The

GTCNT counter is cleared for the channel associated with the GTCLR bit number where 1 is written. Writing 0 has no effect on the status of the GTCNT counter.

CCLRx bit (channel x GTCNT Count Clear) (x = 0 to 7).

Channel x of the GTCNT counter value is cleared on writing 1 to the CCLRx bit. This bit is read as 0.

18.2.5 General PWM Timer Start Source Select Register (GTSSR)

Address(es): GPT32Em.GTSSR H’1004_8010 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CSTRT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSCBFAH	SSCBFAL	SSCBRAH	SSCBRAL	SSCAFBH	SSCAFBL	SSCARBH	SSCARBL	SSGTRGDF	SSGTRGDR	SSGTRGCF	SSGTRGCR	SSGTRGBF	SSGTRGBR	SSGTRGAF	SSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	SSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGA input 1: Enable counter start on the rising edge of GTETRGA input
b1	SSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGA input 1: Enable counter start on the falling edge of GTETRGA input
b2	SSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGB input 1: Enable counter start on the rising edge of GTETRGB input
b3	SSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGB input 1: Enable counter start on the falling edge of GTETRGB input
b4	SSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGC input 1: Enable counter start on the rising edge of GTETRGC input
b5	SSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGC input 1: Enable counter start on the falling edge of GTETRGC input
b6	SSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Start Enable 0: Disable counter start on the rising edge of GTETRGD input 1: Enable counter start on the rising edge of GTETRGD input
b7	SSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Start Enable 0: Disable counter start on the falling edge of GTETRGD input 1: Enable counter start on the falling edge of GTETRGD input
b8	SSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter start on the rising edge of GTIOCA input when GTIOCB input is 0
b9	SSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter start on the rising edge of GTIOCA input when GTIOCB input is 1

Bit	Bit Name	Initial Value	R/W	Description
b10	SSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter start on the falling edge of GTIOCA input when GTIOCB input is 0
b11	SSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter start on the falling edge of GTIOCA input when GTIOCB input is 1
b12	SSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter start on the rising edge of GTIOCB input when GTIOCA input is 0
b13	SSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Start Enable 0: Disable counter start on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter start on the rising edge of GTIOCB input when GTIOCA input is 1
b14	SSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter start on the falling edge of GTIOCB input when GTIOCA input is 0
b15	SSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Start Enable 0: Disable counter start on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter start on the falling edge of GTIOCB input when GTIOCA input is 1
b30 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b31	CSTRT	0	R/W	Software Source Counter Start Enable 0: Disable counter start by the GTSTR register 1: Enable counter start by the GTSTR register

GTSSR sets the source to start the GTCNT counter.

SSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Start Enable)

The SSGTRGAR bit enables or disables GTCNT counter start on the rising edge of GTETRGA pin input.

SSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Start Enable)

The SSGTRGAF bit enables or disables GTCNT counter start on the falling edge of GTETRGA pin input.

SSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Start Enable)

The SSGTRGBR bit enables or disables GTCNT counter start on the rising edge of GTETRGB pin input.

SSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Start Enable)

The SSGTRGBF bit enables or disables GTCNT counter start on the falling edge of GTETRGB pin input.

SSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Start Enable)

The SSGTRGCR bit enables or disables GTCNT counter start on the rising edge of GTETRGC pin input.

SSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Start Enable)

The SSGTRGCF bit enables or disables GTCNT counter start on the falling edge of GTETRGC pin input.

SSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Start Enable)

The SSGTRGDR bit enables or disables GTCNT counter start on the rising edge of GTETRGD pin input.

SSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Start Enable)

The SSGTRGDF bit enables or disables GTCNT counter start on the falling edge of GTETRGD pin input.

SSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Start Enable)

The SSCARBL bit enables or disables GTCNT counter start on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

SSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Start Enable)

The SSCARBH bit enables or disables GTCNT counter start on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

SSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Start Enable)

The SSCAFBL bit enables or disables GTCNT counter start on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

SSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Start Enable)

The SSCAFBH bit enables or disables GTCNT counter start on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

SSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Start Enable)

The SSCBRAL bit enables or disables GTCNT counter start on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

SSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Start Enable)

The SSCBRAH bit enables or disables GTCNT counter start on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

SSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Start Enable)

The SSCBFAL bit enables or disables GTCNT counter start on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

SSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Start Enable)

The SSCBFAH bit enables or disables GTCNT counter start on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

CSTRT bit (Software Source Counter Start Enable)

The CSTRT bit enables or disables GTCNT counter start by GTSTR register.

18.2.6 General PWM Timer Stop Source Select Register (GTPSR)

Address(es): GPT32Em.GTPSR H’1004_8014 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CSTOP	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PSCBFAH	PSCBFAL	PSCBRAH	PSCBRAL	PSCAFBH	PSCAFBL	PSCARBH	PSCARBL	PSGTRGDF	PSGTRGDR	PSGTRGCF	PSGTRGCR	PSGTRGBF	PSGTRGBR	PSGTRGAF	PSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	PSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGA input 1: Enable counter stop on the rising edge of GTETRGA input
b1	PSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGA input 1: Enable counter stop on the falling edge of GTETRGA input
b2	PSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGB input 1: Enable counter stop on the rising edge of GTETRGB input
b3	PSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGB input 1: Enable counter stop on the falling edge of GTETRGB input
b4	PSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGC input 1: Enable counter stop on the rising edge of GTETRGC input
b5	PSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGC input 1: Enable counter stop on the falling edge of GTETRGC input
b6	PSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTETRGD input 1: Enable counter stop on the rising edge of GTETRGD input
b7	PSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTETRGD input 1: Enable counter stop on the falling edge of GTETRGD input
b8	PSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter stop on the rising edge of GTIOCA input when GTIOCB input is 0
b9	PSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter stop on the rising edge of GTIOCA input when GTIOCB input is 1

Bit	Bit Name	Initial Value	R/W	Description
b10	PSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter stop on the falling edge of GTIOCA input when GTIOCB input is 0
b11	PSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter stop on the falling edge of GTIOCA input when GTIOCB input is 1
b12	PSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter stop on the rising edge of GTIOCB input when GTIOCA input is 0
b13	PSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Stop Enable 0: Disable counter stop on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter stop on the rising edge of GTIOCB input when GTIOCA input is 1
b14	PSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter stop on the falling edge of GTIOCB input when GTIOCA input is 0
b15	PSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Stop Enable 0: Disable counter stop on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter stop on the falling edge of GTIOCB input when GTIOCA input is 1
b30 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b31	CSTOP	0	R/W	Software Source Counter Stop Enable 0: Disable counter stop by the GTSTP register 1: Enable counter stop by the GTSTP register

GTPSR sets the source to stop the GTCNT counter.

PSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Stop Enable)

The PSGTRGAR bit enables or disables GTCNT counter stop on the rising edge of GTETRGA pin input.

PSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Stop Enable)

The PSGTRGAF bit enables or disables GTCNT counter stop on the falling edge of GTETRGA pin input.

PSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Stop Enable)

The PSGTRGBR bit enables or disables GTCNT counter stop on the rising edge of GTETRGB pin input.

PSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Stop Enable)

The PSGTRGBF bit enables or disables GTCNT counter stop on the falling edge of GTETRGB pin input.

PSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Stop Enable)

The PSGTRGCR bit enables or disables GTCNT counter stop on the rising edge of GTETRGC pin input.

PSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Stop Enable)

The PSGTRGCF bit enables or disables GTCNT counter stop on the falling edge of GTETRGC pin input.

PSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Stop Enable)

The PSGTRGDR bit enables or disables GTCNT counter stop on the rising edge of GTETRGD pin input.

PSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Stop Enable)

The PSGTRGDF bit enables or disables GTCNT counter stop on the falling edge of GTETRGD pin input.

PSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Stop Enable)

The PSCARBL bit enables or disables GTCNT counter stop on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

PSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Stop Enable)

The PSCARBH bit enables or disables GTCNT counter stop on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

PSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Stop Enable)

The PSCAFBL bit enables or disables GTCNT counter stop on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

PSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Stop Enable)

The PSCAFBH bit enables or disables GTCNT counter stop on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

PSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Stop Enable)

The PSCBRAL bit enables or disables GTCNT counter stop on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

PSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Stop Enable)

The PSCBRAH bit enables or disables GTCNT counter stop on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

PSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Stop Enable)

The PSCBFAL bit enables or disables GTCNT counter stop on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

PSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Stop Enable)

The PSCBFAH bit enables or disables GTCNT counter stop on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

CSTOP bit (Software Source Counter Stop Enable)

The CSTOP bit enables or disables GTCNT counter stop by GTSTP register.

18.2.7 General PWM Timer Clear Source Select Register (GTCSR)

Address(es): GPT32Em.GTCSR H'1004_8018 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CCLR	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CSCBFAH	CSCBFAL	CSCBRAH	CSCBRAL	CSCAFBH	CSCAFBL	CSCARBH	CSCARBL	CSGTRGDF	CSGTRGDR	CSGTRGCF	CSGTRGCR	CSGTRGBF	CSGTRGBR	CSGTRGAF	CSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGA input 1: Enable counter clear on the rising edge of GTETRGA input
b1	CSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGA input 1: Enable counter clear on the falling edge of GTETRGA input
b2	CSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGB input 1: Enable counter clear on the rising edge of GTETRGB input
b3	CSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGB input 1: Enable counter clear on the falling edge of GTETRGB input
b4	CSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGC input 1: Enable counter clear on the rising edge of GTETRGC input
b5	CSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGC input 1: Enable counter clear on the falling edge of GTETRGC input
b6	CSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTETRGD input 1: Enable counter clear on the rising edge of GTETRGD input
b7	CSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTETRGD input 1: Enable counter clear on the falling edge of GTETRGD input
b8	CSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter clear on the rising edge of GTIOCA input when GTIOCB input is 0
b9	CSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter clear on the rising edge of GTIOCA input when GTIOCB input is 1

Bit	Bit Name	Initial Value	R/W	Description
b10	CSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter clear on the falling edge of GTIOCA input when GTIOCB input is 0
b11	CSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter clear on the falling edge of GTIOCA input when GTIOCB input is 1
b12	CSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter clear on the rising edge of GTIOCB input when GTIOCA input is 0
b13	CSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Clear Enable 0: Disable counter clear on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter clear on the rising edge of GTIOCB input when GTIOCA input is 1
b14	CSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter clear on the falling edge of GTIOCB input when GTIOCA input is 0
b15	CSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Clear Enable 0: Disable counter clear on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter clear on the falling edge of GTIOCB input when GTIOCA input is 1
b30 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b31	CCLR	0	R/W	Software Source Counter Clear Enable 0: Disable counter clear by the GTCLR register 1: Enable counter clear by the GTCLR register

GTCSR sets the source to clear the GTCNT counter.

CSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Clear Enable)

The CSGTRGAR bit enables or disables GTCNT counter clear on the rising edge of GTETRGA pin input.

CSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Clear Enable)

The CSGTRGAF bit enables or disables GTCNT counter clear on the falling edge of GTETRGA pin input.

CSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Clear Enable)

The CSGTRGBR bit enables or disables GTCNT counter clear on the rising edge of GTETRGB pin input.

CSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Clear Enable)

The CSGTRGBF bit enables or disables GTCNT counter clear on the falling edge of GTETRGB pin input.

CSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Clear Enable)

The CSGTRGCR bit enables or disables GTCNT counter clear on the rising edge of GTETRGC pin input.

CSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Clear Enable)

The CSGTRGCF bit enables or disables GTCNT counter clear on the falling edge of GTETRGC pin input.

CSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Clear Enable)

The CSGTRGDR bit enables or disables GTCNT counter clear on the rising edge of GTETRGD pin input.

CSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Clear Enable)

The CSGTRGDF bit enables or disables GTCNT counter clear on the falling edge of GTETRGD pin input.

CSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Clear Enable)

The CSCARBL bit enables or disables GTCNT counter clear on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

CSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Clear Enable)

The CSCARBH bit enables or disables GTCNT counter clear on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

CSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Clear Enable)

The CSCAFBL bit enables or disables GTCNT counter clear on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

CSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Clear Enable)

The CSCAFBH bit enables or disables GTCNT counter clear on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

CSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Clear Enable)

The CSCBRAL bit enables or disables GTCNT counter clear on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

CSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Clear Enable)

The CSCBRAH bit enables or disables GTCNT counter clear on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

CSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Clear Enable)

The CSCBFAL bit enables or disables GTCNT counter clear on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

CSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Clear Enable)

The CSCBFAH bit enables or disables GTCNT counter clear on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

CCLR bit (Software Source Counter Clear Enable)

The CCLR bit enables or disables GTCNT counter clear by GTCLR register.

18.2.8 General PWM Timer Up Count Source Select Register (GTUPSR)

Address(es): GPT32Em.GTUPSR H’1004_801C + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	USCBFAH	USCBFAL	USCBRAH	USCBRAL	USCAFBH	USCAFBL	USCARBH	USCARBL	USGTRGDF	USGTRGDR	USGTRGCF	USGTRGCR	USGTRGBF	USGTRGBR	USGTRGAF	USGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	USGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGA input 1: Enable counter count up on the rising edge of GTETRGA input.
b1	USGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGA input 1: Enable counter count up on the falling edge of GTETRGA input.
b2	USGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGB input 1: Enable counter count up on the rising edge of GTETRGB input.
b3	USGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGB input 1: Enable counter count up on the falling edge of GTETRGB input.
b4	USGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGC input 1: Enable counter count up on the rising edge of GTETRGC input.
b5	USGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGC input 1: Enable counter count up on the falling edge of GTETRGC input.
b6	USGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTETRGD input 1: Enable counter count up on the rising edge of GTETRGD input.
b7	USGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTETRGD input 1: Enable counter count up on the falling edge of GTETRGD input.
b8	USCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count up on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	USCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count up on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	USCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count up on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	USCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count up on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	USCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count up on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	USCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Up Enable 0: Disable counter count up on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count up on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	USCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count up on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	USCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Up Enable 0: Disable counter count up on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count up on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTUPSR sets the source to count up the GTCNT counter.

When at least 1 bit in the GTUPSR register is set to 1, the GTCNT counter is counted up by the source that is set to 1 in this register. In such cases, the setting of GTCR.TPCS has no effect.

USGTRGAR bit (GTETRGA Pin Rising Input Source Counter Count Up Enable)

The USGTRGAR bit enables or disables GTCNT counter count up on the rising edge of GTETRGA pin input.

USGTRGAF bit (GTETRGA Pin Falling Input Source Counter Count Up Enable)

The USGTRGAF bit enables or disables GTCNT counter count up on the falling edge of GTETRGA pin input.

USGTRGBR bit (GTETRGB Pin Rising Input Source Counter Count Up Enable)

The USGTRGBR bit enables or disables GTCNT counter count up on the rising edge of GTETRGB pin input.

USGTRGBF bit (GTETRGB Pin Falling Input Source Counter Count Up Enable)

The USGTRGBF bit enables or disables GTCNT counter count up on the falling edge of GTETRGB pin input.

USGTRGCR bit (GTETRGC Pin Rising Input Source Counter Count Up Enable)

The USGTRGCR bit enables or disables GTCNT counter count up on the rising edge of GTETRGC pin input.

USGTRGCF bit (GTETRGC Pin Falling Input Source Counter Count Up Enable)

The USGTRGCF bit enables or disables GTCNT counter count up on the falling edge of GTETRGC pin input.

USGTRGDR bit (GTETRGD Pin Rising Input Source Counter Count Up Enable)

The USGTRGDR bit enables or disables GTCNT counter count up on the rising edge of GTETRGD pin input.

USGTRGDF bit (GTETRGD Pin Falling Input Source Counter Count Up Enable)

The USGTRGDF bit enables or disables GTCNT counter count up on the falling edge of GTETRGD pin input.

USCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Up Enable)

The USCARBL bit enables or disables GTCNT counter count up on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

USCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Up Enable)

The USCARBH bit enables or disables GTCNT counter count up on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

USCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Up Enable)

The USCAFBL bit enables or disables GTCNT counter count up on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

USCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Up Enable)

The USCAFBH bit enables or disables GTCNT counter count up on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

USCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Up Enable)

The USCBRAL bit enables or disables GTCNT counter count up on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

USCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Up Enable)

The USCBRAH bit enables or disables GTCNT counter count up on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

USCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Up Enable)

The USCBFAL bit enables or disables GTCNT counter count up on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

USCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Up Enable)

The USCBFAH bit enables or disables GTCNT counter count up on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.9 General PWM Timer Down Count Source Select Register (GTDNSR)

Address(es): GPT32Em.GTDNSR H’1004_8020 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DSCBFAH	DSCBFAL	DSCBRAH	DSCBRAL	DSCAFBH	DSCAFBL	DSCARBH	DSCARBL	DSGTRGDF	DSGTRGDR	DSGTRGCF	DSGTRGCR	DSGTRGBF	DSGTRGBR	DSGTRGAF	DSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	DSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGA input 1: Enable counter count down on the rising edge of GTETRGA input.
b1	DSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGA input 1: Enable counter count down on the falling edge of GTETRGA input.
b2	DSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGB input 1: Enable counter count down on the rising edge of GTETRGB input.
b3	DSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGB input 1: Enable counter count down on the falling edge of GTETRGB input.
b4	DSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGC input 1: Enable counter count down on the rising edge of GTETRGC input.
b5	DSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGC input 1: Enable counter count down on the falling edge of GTETRGC input.
b6	DSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTETRGD input 1: Enable counter count down on the rising edge of GTETRGD input.
b7	DSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTETRGD input 1: Enable counter count down on the falling edge of GTETRGD input.
b8	DSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count down on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	DSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count down on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	DSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable counter count down on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	DSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable counter count down on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	DSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count down on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	DSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Down Enable 0: Disable counter count down on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count down on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	DSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable counter count down on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	DSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Down Enable 0: Disable counter count down on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable counter count down on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTDNSR sets the source to count down the GTCNT counter.

When at least 1 bit in the GTDNSR register is set to 1, the GTCNT counter is counted down by the source that is set to 1 in this register. In such cases, the setting of GTCR.TPCS has no effect.

DSGTRGAR bit (GTETRGA Pin Rising Input Source Counter Count Down Enable)

The DSGTRGAR bit enables or disables GTCNT counter count down on the rising edge of GTETRGA pin input.

DSGTRGAF bit (GTETRGA Pin Falling Input Source Counter Count Down Enable)

The DSGTRGAF bit enables or disables GTCNT counter count down on the falling edge of GTETRGA pin input.

DSGTRGBR bit (GTETRGB Pin Rising Input Source Counter Count Down Enable)

The DSGTRGBR bit enables or disables GTCNT counter count down on the rising edge of GTETRGB pin input.

DSGTRGBF bit (GTETRGB Pin Falling Input Source Counter Count Down Enable)

The DSGTRGBF bit enables or disables GTCNT counter count down on the falling edge of GTETRGB pin input.

DSGTRGCR bit (GTETRGC Pin Rising Input Source Counter Count Down Enable)

The DSGTRGCR bit enables or disables GTCNT counter count down on the rising edge of GTETRGC pin input.

DSGTRGCF bit (GTETRGC Pin Falling Input Source Counter Count Down Enable)

The DSGTRGCF bit enables or disables GTCNT counter count down on the falling edge of GTETRGC pin input.

DSGTRGDR bit (GTETRGD Pin Rising Input Source Counter Count Down Enable)

The DSGTRGDR bit enables or disables GTCNT counter count down on the rising edge of GTETRGD pin input.

DSGTRGDF bit (GTETRGD Pin Falling Input Source Counter Count Down Enable)

The DSGTRGDF bit enables or disables GTCNT counter count down on the falling edge of GTETRGD pin input.

DSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source Counter Count Down Enable)

The DSCARBL bit enables or disables GTCNT counter count down on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

DSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source Counter Count Down Enable)

The DSCARBH bit enables or disables GTCNT counter count down on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

DSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source Counter Count Down Enable)

The DSCAFBL bit enables or disables GTCNT counter count down on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

DSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source Counter Count Down Enable)

The DSCAFBH bit enables or disables GTCNT counter count down on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

DSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source Counter Count Down Enable)

The DSCBRAL bit enables or disables GTCNT counter count down on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

DSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source Counter Count Down Enable)

The DSCBRAH bit enables or disables GTCNT counter count down on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

DSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source Counter Count Down Enable)

The DSCBFAL bit enables or disables GTCNT counter count down on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

DSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source Counter Count Down Enable)

The DSCBFAH bit enables or disables GTCNT counter count down on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.10 General PWM Timer Input Capture Source Select Register A (GTICASR)

Address(es): GPT32Em.GTICASR H'1004_8024 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ASCBFAH	ASCBFAL	ASCBRAH	ASCBRAL	ASCAFBH	ASCAFBL	ASCARBH	ASCARBL	ASGTRGDF	ASGTRGDR	ASGTRGCF	ASGTRGCR	ASGTRGBF	ASGTRGBR	ASGTRGAF	ASGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ASGTRGAR	0	R/W	GTETRGA Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGA input 1: Enable GTCCRA input capture on the rising edge of GTETRGA input.
b1	ASGTRGAF	0	R/W	GTETRGA Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGA input 1: Enable GTCCRA input capture on the falling edge of GTETRGA input.
b2	ASGTRGBR	0	R/W	GTETRGB Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGB input 1: Enable GTCCRA input capture on the rising edge of GTETRGB input.
b3	ASGTRGBF	0	R/W	GTETRGB Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGB input 1: Enable GTCCRA input capture on the falling edge of GTETRGB input.
b4	ASGTRGCR	0	R/W	GTETRGC Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGC input 1: Enable GTCCRA input capture on the rising edge of GTETRGC input.
b5	ASGTRGCF	0	R/W	GTETRGC Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGC input 1: Enable GTCCRA input capture on the falling edge of GTETRGC input.
b6	ASGTRGDR	0	R/W	GTETRGD Pin Rising Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTETRGD input 1: Enable GTCCRA input capture on the rising edge of GTETRGD input.
b7	ASGTRGDF	0	R/W	GTETRGD Pin Falling Input Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTETRGD input 1: Enable GTCCRA input capture on the falling edge of GTETRGD input.
b8	ASCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	ASCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRA input capture on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	ASCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	ASCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRA input capture on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	ASCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	ASCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRA input capture on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	ASCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	ASCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRA Input Capture Enable 0: Disable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRA input capture on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTICASR sets the source of input capture for GTCCRA.

ASGTRGAR bit (GTETRGA Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGAR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGA pin input.

ASGTRGAF bit (GTETRGA Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGAF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGA pin input.

ASGTRGBR bit (GTETRGB Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGBR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGB pin input.

ASGTRGBF bit (GTETRGB Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGBF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGB pin input.

ASGTRGCR bit (GTETRGC Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGCR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGC pin input.

ASGTRGCF bit (GTETRGC Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGCF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGC pin input.

ASGTRGDR bit (GTETRGD Pin Rising Input Source GTCCRA Input Capture Enable)

The ASGTRGDR bit enables or disables input capture for GTCCRA on the rising edge of GTETRGD pin input.

ASGTRGDF bit (GTETRGD Pin Falling Input Source GTCCRA Input Capture Enable)

The ASGTRGDF bit enables or disables input capture for GTCCRA on the falling edge of GTETRGD pin input.

ASCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRA Input Capture Enable)

The ASCARBL bit enables or disables input capture for GTCCRA on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

ASCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRA Input Capture Enable)

The ASCARBH bit enables or disables input capture for GTCCRA on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

ASCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRA Input Capture Enable)

The ASCAFBL bit enables or disables input capture for GTCCRA on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

ASCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRA Input Capture Enable)

The ASCAFBH bit enables or disables input capture for GTCCRA on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

ASCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRA Input Capture Enable)

The ASCBRAL bit enables or disables input capture for GTCCRA on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

ASCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRA Input Capture Enable)

The ASCBRAH bit enables or disables input capture for GTCCRA on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

ASCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRA Input Capture Enable)

The ASCBFAL bit enables or disables input capture for GTCCRA on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

ASCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRA Input Capture Enable)

The ASCBFAH bit enables or disables input capture for GTCCRA on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.11 General PWM Timer Input Capture Source Select Register B (GTICBSR)

Address(es): GPT32Em.GTICBSR H'1004_8028 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BSCBFAH	BSCBFAL	BSCBRAH	BSCBRAL	BSCAFBH	BSCAFBL	BSCARBH	BSCARBL	BSGTRGDF	BSGTRGDR	BSGTRGCF	BSGTRGCR	BSGTRGBF	BSGTRGBR	BSGTRGAF	BSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	BSGTRGAR	0	R/W	GTETRGA Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGA input 1: Enable GTCCRB input capture on the rising edge of GTETRGA input.
b1	BSGTRGAF	0	R/W	GTETRGA Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGA input 1: Enable GTCCRB input capture on the falling edge of GTETRGA input.
b2	BSGTRGBR	0	R/W	GTETRGB Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGB input 1: Enable GTCCRB input capture on the rising edge of GTETRGB input.
b3	BSGTRGBF	0	R/W	GTETRGB Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGB input 1: Enable GTCCRB input capture on the falling edge of GTETRGB input.
b4	BSGTRGCR	0	R/W	GTETRGC Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGC input 1: Enable GTCCRB input capture on the rising edge of GTETRGC input.
b5	BSGTRGCF	0	R/W	GTETRGC Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGC input 1: Enable GTCCRB input capture on the falling edge of GTETRGC input.
b6	BSGTRGDR	0	R/W	GTETRGD Pin Rising Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTETRGD input 1: Enable GTCCRB input capture on the rising edge of GTETRGD input.
b7	BSGTRGDF	0	R/W	GTETRGD Pin Falling Input Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTETRGD input 1: Enable GTCCRB input capture on the falling edge of GTETRGD input.
b8	BSCARBL	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 0.
b9	BSCARBH	0	R/W	GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRB input capture on the rising edge of GTIOCA input when GTIOCB input is 1.

Bit	Bit Name	Initial Value	R/W	Description
b10	BSCAFBL	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 0 1: Enable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 0.
b11	BSCAFBH	0	R/W	GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 1 1: Enable GTCCRB input capture on the falling edge of GTIOCA input when GTIOCB input is 1.
b12	BSCBRAL	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 0.
b13	BSCBRAH	0	R/W	GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRB input capture on the rising edge of GTIOCB input when GTIOCA input is 1.
b14	BSCBFAL	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 0 1: Enable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 0.
b15	BSCBFAH	0	R/W	GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRB Input Capture Enable 0: Disable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 1 1: Enable GTCCRB input capture on the falling edge of GTIOCB input when GTIOCA input is 1.
b31 to b16	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTICBSR sets the source of input capture for GTCCRB.

BSGTRGAR bit (GTETRGA Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGAR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGA pin input.

BSGTRGAF bit (GTETRGA Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGAF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGA pin input.

BSGTRGBR bit (GTETRGB Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGBR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGB pin input.

BSGTRGBF bit (GTETRGB Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGBF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGB pin input.

BSGTRGCR bit (GTETRGC Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGCR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGC pin input.

BSGTRGCF bit (GTETRGC Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGCF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGC pin input.

BSGTRGDR bit (GTETRGD Pin Rising Input Source GTCCRB Input Capture Enable)

The BSGTRGDR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGD pin input.

BSGTRGDF bit (GTETRGD Pin Falling Input Source GTCCRB Input Capture Enable)

The BSGTRGDF bit enables or disables input capture for GTCCRB on the falling edge of GTETRGD pin input.

BSCARBL bit (GTIOCA Pin Rising Input during GTIOCB Value Low Source GTCCRB Input Capture Enable)

The BSCARBL bit enables or disables input capture for GTCCRB on the rising edge of GTIOCA pin input, when GTIOCB input is 0.

BSCARBH bit (GTIOCA Pin Rising Input during GTIOCB Value High Source GTCCRB Input Capture Enable)

The BSCARBH bit enables or disables input capture for GTCCRB on the rising edge of GTIOCA pin input, when GTIOCB input is 1.

BSCAFBL bit (GTIOCA Pin Falling Input during GTIOCB Value Low Source GTCCRB Input Capture Enable)

The BSCAFBL bit enables or disables input capture for GTCCRB on the falling edge of GTIOCA pin input, when GTIOCB input is 0.

BSCAFBH bit (GTIOCA Pin Falling Input during GTIOCB Value High Source GTCCRB Input Capture Enable)

The BSCAFBH bit enables or disables input capture for GTCCRB on the falling edge of GTIOCA pin input, when GTIOCB input is 1.

BSCBRAL bit (GTIOCB Pin Rising Input during GTIOCA Value Low Source GTCCRB Input Capture Enable)

The BSCBRAL bit enables or disables input capture for GTCCRB on the rising edge of GTIOCB pin input, when GTIOCA input is 0.

BSCBRAH bit (GTIOCB Pin Rising Input during GTIOCA Value High Source GTCCRB Input Capture Enable)

The BSCBRAH bit enables or disables input capture for GTCCRB on the rising edge of GTIOCB pin input, when GTIOCA input is 1.

BSCBFAL bit (GTIOCB Pin Falling Input during GTIOCA Value Low Source GTCCRB Input Capture Enable)

The BSCBFAL bit enables or disables input capture for GTCCRB on the falling edge of GTIOCB pin input, when GTIOCA input is 0.

BSCBFAH bit (GTIOCB Pin Falling Input during GTIOCA Value High Source GTCCRB Input Capture Enable)

The BSCBFAH bit enables or disables input capture for GTCCRB on the falling edge of GTIOCB pin input, when GTIOCA input is 1.

18.2.12 General PWM Timer Control Register (GTCR)

Address(es): GPT32Em.GTCR H'1004_802C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	TPCS[2:0]			—	—	—	—	—	MD[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	CST	0	R/W	Count Start 0: Stop count operation 1: Perform count operation.
b15 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b18 to b16	MD[2:0]	All 0	R/W	Mode Select b18 b16 0 0 0: Saw-wave PWM mode (single buffer or double buffer possible) 0 0 1: Saw-wave one-shot pulse mode (fixed buffer operation) 0 1 0: Setting prohibited 0 1 1: Setting prohibited 1 0 0: Triangle-wave PWM mode 1 (32-bit transfer at trough) (single buffer or double buffer possible) 1 0 1: Triangle-wave PWM mode 2 (32-bit transfer at crest and trough) (single buffer or double buffer possible) 1 1 0: Triangle-wave PWM mode 3 (64-bit transfer at trough) fixed buffer operation) 1 1 1: Setting prohibited.
b23 to b19	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b26 to b24	TPCS[2:0]	All 0	R/W	Timer Prescaler Select b26 b24 0 0 0: P0ϕ 0 0 1: P0ϕ/4 0 1 0: P0ϕ/16 0 1 1: P0ϕ/64 1 0 0: P0ϕ/256 1 0 1: P0ϕ/1024.
b31 to b27	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTCR controls GTCNT.

CST bit (Count Start)

The CST bit controls the GTCNT counter start and stop.

[Setting conditions]

- GTSTR value where the channel number associated with the bit number is set to 1 with the GTSSR.CSTRT bit being 1.
- among input on the GTIOCA, GTIOCB, or GTETRGA to GTETRGD pins that is enabled in the GTSSR register as a source for the start of counting
- 1 is written by software directly.

[Clearing conditions]

- GTSTP value where the channel number associated with the bit number is set to 1 with the GTPSR.CSTOP bit being 1.
- among input on the GTIOCA, GTIOCB, or GTETRGA to GTETRGD pins that is enabled in the GTPSR register as a source for the stop of counting
- 0 is written by software directly.

MD[2:0] bits (Mode Select)

The MD[2:0] bits select the GPT operating mode. The MD[2:0] bits must be set while the GTCNT operation is stopped.

TPCS[2:0] bits (Timer Prescaler Select)

The TPCS[2:0] bits select the clock for GTCNT. A clock prescaler can be selected independently for each channel. The TPCS[2:0] bits must be set while the GTCNT operation is stopped.

18.2.13 General PWM Timer Count Direction and Duty Setting Register (GTUDDTYC)

Address(es): GPT32Em.GTUDDTYC H’1004_8030 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	OBDTYR	OBDTYF	OBDTY[1:0]		—	—	—	—	OADTYR	OADTYF	OADTY[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UDF	UD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	UD	1	R/W	Count Direction Setting 0: Count down on GTCNT 1: Counts up on GTCNT.
b1	UDF	0	R/W	Forcible Count Direction Setting 0: Do not force setting 1: Force setting.
b15 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b17, b16	OADTY[1:0]	All 0	R/W	GTIOCA Output Duty Setting b17 b16 0 x: GTIOCA pin duty depends on compare match 1 0: GTIOCA pin duty 0% 1 1: GTIOCA pin duty 100%.
b18	OADTYF	0	R/W	Forcible GTIOCA Output Duty Setting 0: Do not force setting 1: Force setting.
b19	OADTYR	0	R/W	GTIOCA Output Value Selecting after Releasing 0%/100% Duty Setting 0: Apply output value set in 0%/100% duty to GTIOA[3:2] function after releasing 0%/100% duty setting 1: Apply masked compare match output value to GTIOA[3:2] function after releasing 0%/100% duty setting.
b23 to b20	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b25, b24	OBDTY[1:0]	All 0	R/W	GTIOCB Output Duty Setting b25 b24 0 x: GTIOCB pin duty depends on compare match 1 0: GTIOCB pin duty = 0% 1 1: GTIOCB pin duty = 100%.
b26	OBDTYF	0	R/W	Forcible GTIOCB Output Duty Setting 0: Do not force setting 1: Force setting.
b27	OBDTYR	0	R/W	GTIOCB Output Value Selecting after Releasing 0%/100% Duty Setting 0: Apply output value set in 0%/100% duty to GTIOB[3:2] function after releasing 0%/100% duty setting 1: Apply masked compare match output value to GTIOB[3:2] function after releasing 0%/100% duty setting.

Bit	Bit Name	Initial Value	R/W	Description
b31 to b28	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Remarks: x: Don't care

GTUDDTYC sets the direction in which GTCNT counts (up-counting or down-counting) and sets the duty of GTIOCA/ GTIOCB pin output.

Count direction:

- In saw-wave mode

  When the UD value is set to 0 during up-counting, the count direction changes at an overflow (the timing synchronous with count clock after the GTCNT value becomes the GTPR value). When the UD value is set to 1 during down- counting, the count direction changes at an underflow (the timing synchronous with count clock after the GTCNT value becomes 0).

  When the UD value changes from 1 to 0 with the UDF bit being 0 and while counting is stopped, the counter starts up- counting and the count direction changes at an overflow (the timing synchronous with count clock after the GTCNT value becomes GTPR value). When the UD value changes from 0 to 1 with the UDF bit being 0 and while counting stops, the counter starts down-counting and the count direction changes at an underflow (the timing synchronous with count clock after the GTCNT value becomes 0).

  When the UDF bit is set to 1 while counting stops, the UD bit value is reflected in the count direction when counting starts.

- In triangle-wave mode

  When the UD value changes during counting, the count direction does not change. When the UD value changes while the UDF bit is 0 and counting stops, the change is not reflected in the count direction when counting starts.

  When the UDF bit is set to 1 while counting stops, the UD value is reflected in the count direction when counting starts.

UD bit (Count Direction Setting)

The UD bit sets the count direction for up-counting or down-counting, for GTCNT.

UDF bit (Forcible Count Direction Setting)

The UDF bit forcibly sets the count direction when GTCNT starts operation as the UD value. Only write 0 to this bit during counter operation. When 1 is written to this bit while counting stops, return this bit to 0 before counting starts.

Output duty:

- In saw-wave mode

  When the OADTY/OBDTY value changes during up-counting, the duty is reflected at an overflow (GTCNT = GTPR). When the OADTY/OBDTY value changes during down-counting, the duty is reflected at an underflow (GTCNT = 0). When the OADTY/OBDTY value changes to 1 with the OADTYF/OBDTYF bit being 0 and while counting is stopped, the output duty is not reflected at starting counter operation. When the count direction is up, the output duty is reflected at an overflow (GTCNT = GTPR). When the count direction is down, the output duty is reflected at an underflow (GTCNT = 0). When the OADTY/OBDTY value changes to 0 with the OADTYF/OBDTYF bit being 1 and while counting stops, the output duty is reflected at the starting counter operation.

- In triangle-wave mode

  When the OADTY/OBDTY value changes during counting, the duty is reflected at an underflow. When the OADTY/OBDTY value changes to 1 with the OADTYF/OBDTYF bit being 0 and while counting stops, the output duty is not reflected at the starting counter operation, however the output duty is reflected at an underflow.

  When the OADTY/OBDTY value changes to 0 with the OADTYF/OBDTYF bit being 1 and while counting stops, the output duty is reflected at starting counter operation.

OmDTY[1:0] bits (GTIOCm Output Duty Setting) (m = A, B)

The OmDTY[1:0] bits set the output duty (0%, 100% or compare match control) of the GTIOCm pin.

OmDTYF bit (Forcible GTIOCm Output Duty Setting) (m = A, B)

The OmDTYF bit forcibly sets the output duty cycle to the OmDTY setting. Set this bit to 0 during counter operation. When this bit is set to 1 while counting is stopped, return this bit to 0 until the first period ends after the counter starts.

OmDTYR bit (GTIOCm Output Value Selecting after Releasing 0%/100% Duty Setting) (m = A, B)

The OmDTYR bits select the value that is the object of output retained or toggled at cycle end, when the control changes from 0%/100% duty setting to compare match for GTIOCm pin and GTIOR.GTIOm[3:2] are set to 00b (output retained at cycle end) or GTIOR.GTIOm[3:2] are set to 11b (output toggled at cycle end).

GPT32 internally continues the compare match operation in performing 0%/100% duty operation. When the OmDTYR bit is set to 1, the GTCIOm pin outputs the value specified in the GTIOR.GTIOm[3:2] bits at the end of compare match cycle.

18.2.14 General PWM Timer I/O Control Register (GTIOR)

Address(es): GPT32Em.GTIOR H’1004_8034 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NFCSB[1:0]		NFBEN	—	—	OBDF[1:0]		OBE	OBHLD	OBDFLT	—	GTIOB[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	NFCSA[1:0]		NFAEN	—	—	OADF[1:0]		OAE	OAHLD	OADFLT	—	GTIOA[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	GTIOA[4:0]	All 0	R/W	GTIOCA Pin Function Select See **Table 18.5**.
b5	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b6	OADFLT	0	R/W	GTIOCA Pin Output Value Setting at the Count Stop 0: Output low on GTIOCA pin when counting stops 1: Output high on GTIOCA pin when counting stops.
b7	OAHLD	0	R/W	GTIOCA Pin Output Setting at the Start/Stop Count 0: Set GTIOCA pin output level on counting start and stop based on the register setting. 1: Retain GTIOCA pin output level on counting start and stop.
b8	OAE	0	R/W	GTIOCA Pin Output Enable 0: Disable output 1: Enable output.
b10, b9	OADF[1:0]	All 0	R/W	GTIOCA Pin Disable Value Setting b10 b9 0 0: Prohibit output disable 0 1: Set GTIOCA pin to Hi-Z on output disable 1 0: Set GTIOCA pin to 0 on output disable 1 1: Set GTIOCA pin to 1 on output disable.
b12, b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b13	NFAEN	0	R/W	Noise Filter A Enable 0: Disable noise filter for GTIOCA pin 1: Enable noise filter for GTIOCA pin.
b15, b14	NFCSA[1:0]	All 0	R/W	Noise Filter A Sampling Clock Select b15 b14 0 0: P0ϕ 0 1: P0ϕ/4 1 0: P0ϕ/16 1 1: P0ϕ/64.
b20 to b16	GTIOB[4:0]	All 0	R/W	GTIOCB Pin Function Select See **Table 18.5**.
b21	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b22	OBDFLT	0	R/W	GTIOCB Pin Output Value Setting at the Count Stop 0: Output low on GTIOCB pin when counting stops 1: Output high on GTIOCB pin when counting stops.
b23	OBHLD	0	R/W	GTIOCB Pin Output Setting at the Start/Stop Count 0: Set GTIOCB pin output level on counting start and stop based on the register setting 1: Retain GTIOCB pin output level on counting start and stop.
b24	OBE	0	R/W	GTIOCB Pin Output Enable 0: Disable output 1: Enable output.
b26, b25	OBDF[1:0]	All 0	R/W	GTIOCB Pin Disable Value Setting b26 b25 0 0: Prohibit output disable 0 1: Set GTIOCB pin to Hi-Z on output disable 1 0: Set GTIOCB pin to 0 on output disable 1 1: Set GTIOCB pin to 1 on output disable.
b28, b27	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b29	NFBEN	0	R/W	Noise Filter B Enable 0: Disable noise filter for GTIOCB pin 1: Enable noise filter for GTIOCB pin.
b31, b30	NFCSB[1:0]	All 0	R/W	Noise Filter B Sampling Clock Select b31 b30 0 0: P0ϕ 0 1: P0ϕ/4 1 0: P0ϕ/16 1 1: P0ϕ/64.

GTIOR sets the functions of the GTIOCA and GTIOCB pins.

GTIOA[4:0] bits (GTIOCA Pin Function Select)

The GTIOA[4:0] bits select the GTIOCA pin function. For details, see **Table 18.5**

OADFLT bit (GTIOCA Pin Output Value Setting at the Count Stop)

The OADFLT bit sets whether the GTIOCA pin outputs high or low when counting stops.

OAHLD bit (GTIOCA Pin Output Setting at the Start/Stop Count)

The OAHLD bit specifies whether the GTIOCA pin output level is retained or the level at the start/stop of counting depends on the register setting.

[When the OAHLD bit is set to 0]

- The value specified in bit [4] of the GTIOA[4:0] bits is output when counting starts
- The value specified in the OADFLT bit is output when counting stops
- If the OADFLT bit is modified while counting stops, it is immediately reflected in the output.

[When the OAHLD bit is set to 1]

- The output is retained when counting starts or stops.

OAE bit (GTIOCA Pin Output Enable)

The OAE bit disables or enables the GTIOCA pin output.

When the GTCCRA register is in use as an input capture register (at least one bit in the GTICASR register is set to 1), signals will not be output from the GTIOCA pin regardless of the setting of this bit.

OADF[1:0] bits (GTIOCA Pin Disable Value Setting)

The OADF bits select a value to be output from the GTIOCA pin in response to the output disable requests.

NFAEN bit (Noise Filter A Enable)

The NFAEN bit disables or enables the noise filter for input from the GTIOCA pin. Because changing the value of the bit might lead to internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSA[1:0] bits (Noise Filter A Sampling Clock Select)

The NFCSA[1:0] bits set the sampling interval for the noise filter of the GTIOCA pin. When setting these bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

GTIOB[4:0] bits (GTIOCB Pin Function Select)

The GTIOB[4:0] bits select the GTIOCB pin function. For details, see **Table 18.5**

OBDFLT bit (GTIOCB Pin Output Value Setting at the Count Stop)

The OBDFLT bit sets whether the GTIOCB pin outputs high or low when counting stops.

OBHLD bit (GTIOCB Pin Output Setting at the Start/Stop Count)

The OBHLD bit specifies whether the GTIOCB pin output level is retained or the level at the start/stop of counting depends on the register setting.

[When the OBHLD bit is set to 0]

- The value specified in bit [4] of the GTIOB[4:0] bits is output when counting starts
- The value specified in the OBDFLT bit is output when counting stops
- If the OBDFLT bit is modified while counting stops, it is immediately reflected in the output.

[When the OBHLD bit is set to 1]

- The output is retained when counting starts or stops.

OBE bit (GTIOCB Pin Output Enable)

The OBE bit disables or enables the GTIOCB pin output.

When the GTCCRB register is in use as an input capture register (at least one bit in the GTICBSR register is set to 1), signals will not be output from the GTIOCB pin regardless of the setting of this bit.

OBDF[1:0] bits (GTIOCB Pin Disable Value Setting)

The OBDF bits select a value to be output from the GTIOCB pin in response to the output disable requests.

NFBEN bit (Noise Filter B Enable)

The NFBEN bit disables or enables the noise filter for input from the GTIOCB pin. Because changing the value of the bit might lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSB[1:0] bits (Noise Filter B Sampling Clock Select)

The NFCSB[1:0] bits set the sampling interval for the noise filter of the GTIOCB pin. When setting these bits, wait for two cycles of the selected sampling interval before setting the input-capture function.

Table 18.5 Settings of GTIOA[4:0] and GTIOB[4:0] bits

GTIOA/GTIOB[4:0] bits					Function		
b4	b3	b2	b1	b0	b4	b3, b2	b1, b0
0	0	0	0	0	Set initial output low	Retain output at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	0	0	0	1			Output low at GTCCRA/GTCCRB compare match
0	0	0	1	0			Output high at GTCCRA/GTCCRB compare match
0	0	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	0	1	0	0		Output low at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	0	1	0	1			Output low at GTCCRA/GTCCRB compare match
0	0	1	1	0			Output high at GTCCRA/GTCCRB compare match
0	0	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	1	0	0	0		Output high at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	1	0	0	1			Output low at GTCCRA/GTCCRB compare match
0	1	0	1	0			Output high at GTCCRA/GTCCRB compare match
0	1	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
0	1	1	0	0		Toggle output at cycle end	Retain output at GTCCRA/GTCCRB compare match
0	1	1	0	1			Output low at GTCCRA/GTCCRB compare match
0	1	1	1	0			Output high at GTCCRA/GTCCRB compare match
0	1	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	0	0	0	0	Set initial output high	Retain output at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	0	0	0	1			Output low at GTCCRA/GTCCRB compare match
1	0	0	1	0			Output high at GTCCRA/GTCCRB compare match
1	0	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	0	1	0	0		Output low at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	0	1	0	1			Output low at GTCCRA/GTCCRB compare match
1	0	1	1	0			Output high at GTCCRA/GTCCRB compare match
1	0	1	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	1	0	0	0		Output high at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	1	0	0	1			Output low at GTCCRA/GTCCRB compare match
1	1	0	1	0			Output high at GTCCRA/GTCCRB compare match
1	1	0	1	1			Toggle output at GTCCRA/GTCCRB compare match
1	1	1	0	0		Toggle output at cycle end	Retain output at GTCCRA/GTCCRB compare match
1	1	1	0	1			Output low at GTCCRA/GTCCRB compare match
1	1	1	1	0			Output high at GTCCRA/GTCCRB compare match
1	1	1	1	1			Toggle output at GTCCRA/GTCCRB compare match

Note 1. The cycle end means an overflow (GTCNT is changed from GTPR to 0 in up-counting) or underflow (GTCNT is changed from 0 to GTPR in down-counting). In this case, the GTCNT counter is cleared for saw waves and for the trough (GTCNT is changed from 0 to 1) for triangle waves.

Note 2. When the timing of a cycle end and the timing of a GTCCRA/GTCCRB compare match are the same in a compare-match operation, the b3 and b2 settings are given priority in saw-wave PWM mode, and the b1 and b0 settings are given priority in any other mode.

Note 3. In event count operation where at least 1 bit in GTUPSR or GTDNSR is set to 1, the setting of b3 and b2 is ignored.

18.2.15 General PWM Timer Interrupt Output Setting Register (GTINTAD)

Address(es): GPT32Em.GTINTAD H’1004_8038 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	GRPABL	GRPABH	GRPDTE	—	—	GRP[1:0]		—	—	—	—	ADTRBDEN	ADTRBUEN	ADTRADEN	ADTRAUEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	GTINTPR[1:0]		GTINTF	GTINTE	GTINTD	GTINTC	GTINTB	GTINTA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	GTINTA	0	R/W	GTCCRA Compare Match/Input Capture Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b1	GTINTB	0	R/W	GTCCRB Compare Match/Input Capture Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b2	GTINTC	0	R/W	GTCCRC Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b3	GTINTD	0	R/W	GTCCRD Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b4	GTINTE	0	R/W	GTCCRE Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b5	GTINTF	0	R/W	GTCCRF Compare Match Interrupt Enable 0: Disable Interrupt request 1: Enable Interrupt request.
b7, b6	GTINTPR[1:0]	All 0	R/W	GTPR Compare Match Interrupt Enable b7 b6 0 0: Disable Interrupt request 0 1: Interrupt requests in response to overflows for saw waves and crests for triangle waves are enabled. 1 0: Interrupt requests in response to underflows for saw waves and troughs for triangle waves are enabled. 1 1: Interrupt requests in response to overflows and underflows for saw waves and crests and troughs for triangle waves are enabled.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b16	ADTRAUEN	0	R/W	GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.
b17	ADTRADEN	0	R/W	GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.

Bit	Bit Name	Initial Value	R/W	Description
b18	ADTRBUEN	0	R/W	GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.
b19	ADTRBDEN	0	R/W	GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable 0: Disable A/D converter start request 1: Enable A/D converter start request.
b23 to b20	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b25, b24	GRP[1:0]	All 0	R/W	Output Disable Source Select b25 b24 0 0: Select Group A output disable request 0 1: Select Group B output disable request 1 0: Select Group C output disable request 1 1: Select Group D output disable request.
b27, b26	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b28	GRPDTE	0	R/W	Dead Time Error Output Disable Request Enable 0: Disable dead time error output disable request 1: Enable dead time error output disable request.
b29	GRPABH	0	R/W	Same Time Output Level High Disable Request Enable 0: Disable same time output level high disable request 1: Enable same time output level high disable request.
b30	GRPABL	0	R/W	Same Time Output Level Low Disable Request Enable 0: Disable same time output level low disable request 1: Enable same time output level low disable request.
b31	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

GTINTAD enables or disables interrupt requests, A/D converter start requests and output disable requests.

GTINTA bit (GTCCRA Compare Match/Input Capture Interrupt Enable)

The GTINTA bit enables and disables the interrupt request (CCMPAn) in response to compare matches with or input capture by the GTCCRA register.

GTINTB bit (GTCCRB Compare Match/Input Capture Interrupt Enable)

The GTINTB bit enables and disables the interrupt request (CCMPBn) in response to compare matches with or input capture by the GTCCRB register.

GTINTC bit (GTCCRC Compare Match Interrupt Enable)

The GTINTC bit enables and disables the interrupt request (CMPCn) in response to compare matches with the GTCCRC register.

GTINTD bit (GTCCRD Compare Match Interrupt Enable)

The GTINTD bit enables and disables the interrupt request (CMPDn) in response to compare matches with the GTCCRD register.

GTINTE bit (GTCCRE Compare Match Interrupt Enable)

The GTINTE bit enables and disables the interrupt request (CMPEn) in response to compare matches with the GTCCRE register.

GTINTF bit (GTCCRF Compare Match Interrupt Enable)

The GTINTF bit enables and disables the interrupt request (CMPFn) in response to compare matches with the GTCCRF register.

GTINTPR[1:0] bits (GTPR Compare Match Interrupt Enable)

The GTINTPR[1:0] bits enable and disable the interrupt requests (OVFn and UNFn) in response to compare matches with the GTPR register (and overflows of the GTCNT counter) and underflows of the GTCNT counter.

ADTRAUEN bit (GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRAUEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT up-counting.

ADTRADEN bit (GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRADEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT down-counting.

ADTRBUEN bit (GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRBUEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT up-counting.

ADTRBDEN bit (GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRBDEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT down-counting.

GRP[1:0] bits (Output Disable Source Select)

The GRP[1:0] bits select GTIOCA pin and GTIOCB pin output disable source. The output disable requests to the POEG are output to the group selected by the GRP[1:0] bits. Each of the output disable sources, a dead time error, simultaneous output of high-level, and low-level signal, is also output in accord with the setting of its enabling bit. GTST.ODF shows the request of output disable source group that is selected with the GRP[1:0] bits. Set the GRP[1:0] bits when GTIOR.OAE and OBE bits are both 0.

GRPDTE bit (Dead Time Error Output Disable Request Enable)

The GRPDTE bit enables or disables dead time error output disable request.

GRPABH bit (Same Time Output Level High Disable Request Enable)

The GRPABH bit enables or disables output disable request when GTIOCA pin and GTIOCB pin output 1 at the same time.

GRPABL bit (Same Time Output Level Low Disable Request Enable)

The GRPABL bit enables or disables output disable request when GTIOCA pin and GTIOCB pin output 0 at the same time.

18.2.16 General PWM Timer Status Register (GTST)

Address(es): GPT32Em.GTST H'1004_803C + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	OABLF	OABHF	DTEF	—	—	—	ODF	—	—	—	—	ADTRBDF	ADTRBUF	ADTRADF	ADTRAUF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TUCF	—	—	—	—	ITCNT[2:0]			TCFPU	TCFPO	TCFF	TCFE	TCFD	TCFC	TCFB	TCFA
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R	R	R	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

Bit	Bit Name	Initial Value	R/W	Description
b0	TCFA	0	R/(W)*1	Input Capture/Compare Match Flag A 0: No input capture/compare match of GTCCRA occurred 1: Input capture/compare match of GTCCRA occurred.
b1	TCFB	0	R/(W)*1	Input Capture/Compare Match Flag B 0: No input capture/compare match of GTCCRB occurred 1: Input capture/compare match of GTCCRB occurred.
b2	TCFC	0	R/(W)*1	Compare Match Flag C 0: No compare match of GTCCRC occurred 1: Compare match of GTCCRC occurred.
b3	TCFD	0	R/(W)*1	Compare Match Flag D 0: No compare match of GTCCRD occurred 1: Compare match of GTCCRD occurred.
b4	TCFE	0	R/(W)*1	Compare Match Flag E 0: No compare match of GTCCRE occurred 1: Compare match of GTCCRE occurred.
b5	TCFF	0	R/(W)*1	Compare Match Flag F 0: No compare match of GTCCRF occurred 1: Compare match of GTCCRF occurred.
b6	TCFPO	0	R/(W)*1	Overflow Flag 0: No overflow (crest) occurred 1: Overflow (crest) occurred.
b7	TCFPU	0	R/(W)*1	Underflow Flag 0: No underflow (trough) occurred 1: Underflow (trough) occurred.
b10 to b8	ITCNT[2:0]	All 0	R	OVFn/UNFn Interrupt Skipping Count Counter Counter for counting the number of times a timer interrupt is skipped.
b14 to b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	TUCF	1	R	Count Direction Flag 0: GTCNT counter is counting down 1: GTCNT counter is counting up.
b16	ADTRAUF	0	R/(W)*1	GTADTRA Compare Match (Counting Up) A/D Converter Start Request Flag 0: A compare match with the GTADTRA register has not occurred during counting up. 1: A compare match with the GTADTRA register has occurred during counting up.

Bit	Bit Name	Initial Value	R/W	Description
b17	ADTRADF	0	R/(W) *1	GTADTRA Compare Match (Counting Down) A/D Converter Start Request Flag 0: A compare match with the GTADTRA register has not occurred during counting down. 1: A compare match with the GTADTRA register has occurred during counting down.
b18	ADTRBUF	0	R/(W) *1	GTADTRB Compare Match (Counting Up) A/D Converter Start Request Flag 0: A compare match with the GTADTRB register has not occurred during counting up. 1: A compare match with the GTADTRB register has occurred during counting up.
b19	ADTRBDF	0	R/(W) *1	GTADTRB Compare Match (Counting Down) A/D Converter Start Request Flag 0: A compare match with the GTADTRB register has not occurred during counting down. 1: A compare match with the GTADTRB register has occurred during counting down.
b23 to b20	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b24	ODF	0	R	Output Disable Flag 0: No output disable request occurred 1: Output disable request occurred.
b27 to b25	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b28	DTEF	0	R	Dead Time Error Flag 0: No dead time error occurred 1: Dead time error occurred.
b29	OABHF	0	R	Same Time Output Level High Flag 0: GTIOCA pin and GTIOCB pin did not output 1 at the same time 1: GTIOCA pin and GTIOCB pin output 1 at the same time.
b30	OABLF	0	R	Same Time Output Level Low Flag 0: GTIOCA pin and GTIOCB pin did not output 0 at the same time 1: GTIOCA pin and GTIOCB pin output 0 at the same time.
b31	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Note 1. Only 0 can be written to this bit. Do not write 1.

GTST indicates the status of the GPT.

TCFA flag (Input Capture/Compare Match Flag A)

The TCFS flag indicates the status for the input capture or compare match of GTCCRA.

[Setting conditions]

- GTCNT = GTCCRA, when the GTCCRA register functions as a compare match register
- GTCNT counter value is transferred to GTCCRA by the input capture signal when the GTCCRA register functions as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFB flag (Input Capture/Compare Match Flag B)

The TCFB flag indicates the status for the input capture or compare match of GTCCRB.

[Setting conditions]

- GTCNT = GTCCRB, when the GTCCRB register functions as a compare match register
- GTCNT counter value is transferred to GTCCRB by the input capture signal when the GTCCRB register function as an input capture register.

[Clearing condition]

- 0 is written to this flag.

TCFC flag (Compare Match Flag C)

The TCFC flag indicates the status for the compare match of GTCCRC.

[Setting condition]

- GTCNT = GTCCRC

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (GTCCRC performs buffer operation).

TCFD flag (Compare Match Flag D)

The TCFD flag indicates the status for the compare match of GTCCRD.

[Setting condition]

- GTCNT = GTCCRD.

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (GTCCRD performs buffer operation).

TCFE flag (Compare Match Flag E)

The TCFE flag indicates the status for the compare match of GTCCRE.

[Setting condition]

- GTCNT = GTCCRE.

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (GTCCRE performs buffer operation).

TCFF flag (Compare Match Flag F)

The TCFF flag indicates the status for the compare match of GTCCRF.

[Setting condition]

- GTCNT = GTCCRF.

[Clearing condition]

- 0 is written to this flag. [Not comparing condition]
- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (Triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (GTCCRF performs buffer operation).

TCFPO flag (Overflow Flag)

The TCFPO flag indicates when an overflow or a crest has occurred.

[Setting conditions]

- In saw-wave mode, an overflow (GTCNT changes from GTPR to 0 in up-counting) has occurred
- In triangle-wave mode, a crest (GTCNT changes from GTPR to GTPR-1) has occurred
- In counting by hardware sources, an overflow (GTCNT changes from GTPR to 0 in up-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

TCFPU flag (Underflow Flag)

The TCFPU flag indicates when an underflow or a trough has occurred.

[Setting conditions]

- In saw-wave mode, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred
- In triangle-wave mode, a crest (GTCNT changes from 0 to 1) has occurred
- In counting by hardware sources, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred.

[Clearing condition]

- 0 is written to this flag.

ITCNT[2:0] bits (OVFn/UNFn Interrupt Skipping Count Counter)

When the OVFn/UNFn (n=0 to 7) interrupt skipping function is used (the GTITC.IVTC[1:0] bits are set to a value other than 00b), the counter in the ITCNT[2:0] bits increments by 1 every time the OVFn/UNFn interrupt source that is selected in GTITC.IVTC[1:0] is generated.

[Clearing conditions]

- The OVFn/UNFn interrupt skipping function is not used (GTITC.IVTT[2:0] is 000b when GTITC.IVTC[1:0] is 00b)
- The OVFn/UNFn interrupt skipping count matches the specified count (ITCNT[2:0] matches the skipping count specified in GTITC.IVTT[2:0]).

TUCF flag (Count Direction Flag)

The TUCF flag indicates the count direction of GTCNT. In event count operation, this flag is set to 1 in up-counting and is set to 0 in down-counting.

ADTRAUF flag (GTADTRA Compare Match (Counting Up) A/D Converter Start Request Flag)

The ADTRAUF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRA during counting up

[Clearing condition]

- 0 is written to this flag.

ADTRADF flag (GTADTRA Compare Match (Counting Down) A/D Converter Start Request Flag)

The ADTRADF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRA during counting down

[Clearing condition]

- 0 is written to this flag.

ADTRBUF flag (GTADTRB Compare Match (Counting Up) A/D Converter Start Request Flag)

The ADTRBUF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRB during counting up

[Clearing condition]

- 0 is written to this flag.

ADTRBDF flag (GTADTRB Compare Match (Counting Down) A/D Converter Start Request Flag)

The ADTRBDF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRB during counting down

[Clearing condition]

- 0 is written to this flag.

ODF flag (Output Disable Flag)

The ODF flag shows the request of the output disable source group that is selected by GRP[1:0] bits. When output is disabled, an output disable control is not released within the same cycle in which an output disable request is negated. It is released in the next cycle.

DTEF flag (Dead Time Error Flag)

The DTEF flag indicates that the timer output toggle point after the automatic addition of dead time has exceeded the timer cycle.

This flag returns to 0 when the timer output toggle point after the automatic addition of dead time is returned to the cycle. This flag is read only. Writing 0 to clear the flag is not allowed.

When an interrupt by the DTEF flag is enabled (GTINTAD.GRPDTE = 1), the DTEF flag is output to POEG as an output disable request each time the DTEF flag changes from 0 to 1.

[Setting condition]

- The timer output toggle point after the automatic addition of dead time has exceeded the timer cycle.
  For triangle wave in up-counting: $GTCCRA - GTDVU \leq 0$
  For triangle wave in down-counting: $GTCCRA - GTDVD < 0$
  For saw wave 1 shot pulse mode in up-counting:
    $GTCCRA - GTDVU < 0$ or $GTCCRA + GTDVD > GTPR$
  For saw wave 1 shot pulse mode in down-counting:
    $GTCCRA + GTDVU > GTPR$ or $GTCCRA - GTDVD < 0$

[Clearing condition]

- The timer output toggle point after the automatic addition of dead time is within the timer cycle.

OABHF flag (Same Time Output Level High Flag)

The OABHF flag indicates that GTIOCA pin and GTIOCB pin output 1 at the same time.

When GTIOCA pin or GTIOCB pin output 0, this flag returns to 0. This flag is read only. Writing 0 to clear the flag is not allowed. When an interrupt by the OABHF flag is enabled (GTINTAD.GRPABH = 1), the OABHF flag is output to POEG as an output disable request.

[Setting condition]

- GTIOCA pin and GTIOCB pin output 1 at the same time when both the OAE and OBE bits are set to 1.

[Clearing conditions]

- GTIOCA pin output value is different from GTIOCB pin output value when both the OAE and OBE bits are set to 1
- GTIOCA pin and GTIOCB pin output 0 at the same time when both the OAE and OBE bits are set to 1
- Either OAE bit or OBE bit is set to 0.

OABLF flag (Same Time Output Level Low Flag)

The OABLF flag indicates that GTIOCA pin and GTIOCB pin output 0 at the same time.

When GTIOCA pin or GTIOCB pin output 1, this flag returns to 0. This flag is read only. Writing 0 to clear the flag is not allowed. When an interrupt by the OABLF flag is enabled (GTINTAD.GRPABL = 1), the OABLF flag is output to POEG as an output disable request.

[Setting condition]

- GTIOCA pin and GTIOCB pin output 0 at the same time when both the OAE and OBE bits are set to 1.

[Clearing conditions]

- GTIOCA pin output value is different from GTIOCB pin output value when both the OAE and OBE bits are set to 1
- GTIOCA pin and GTIOCB pin output 1 at the same time when both the OAE and OBE bits are set to 1
- At least either OAE bit or OBE bit is set to 0.

The compare-target signals to generate the OABHF/OABLF flag are the compare match outputs (PWM outputs) signals before masked by the output disable function. When the output disable state is performed, a compare match also performs continuously in the GPT and the OABHF/OABLF flag is updated in association with the result of the compared values.

18.2.17 General PWM Timer Buffer Enable Register (GTBER)

Address(es): GPT32Em.GTBER H'1004_8040 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	ADTDB	ADTTB[1:0]		—	ADTDA	ADTTA[1:0]		—	CCRSWT	PR[1:0]		CCRB[1:0]		CCRA[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	BD[3]	BD[2]	BD[1]	BD[0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	BD[0]	0	R/W	GTCCR Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b1	BD[1]	0	R/W	GTPR Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b2	BD[2]	0	R/W	GTADTR Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b3	BD[3]	0	R/W	GTDV Buffer Operation Disable 0: Enable buffer operation 1: Disable buffer operation.
b15 to b4	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b17, b16	CCRA[1:0]	All 0	R/W	GTCCRA Buffer Operation b17 b16 0 0: No buffer operation 0 1: Single buffer operation (GTCCRA ⇔ GTCCRC) 1 x: Double buffer operation (GTCCRA ⇔ GTCCRC ⇔ GTCCRD).
b19, b18	CCRB[1:0]	All 0	R/W	GTCCRB Buffer Operation b19 b18 0 0: No buffer operation 0 1: Single buffer operation (GTCCRB ⇔ GTCCRE) 1 x: Double buffer operation (GTCCRB ⇔ GTCCRE ⇔ GTCCRF).
b21, b20	PR[1:0]	All 0	R/W	GTPR Buffer Operation b21 b20 0 0: No buffer operation 0 1: Single buffer operation (GTPBR ⇔ GTPR) 1 x: Double buffer operation (GTPDBR ⇔ GTPBR ⇔ GTPR).
b22	CCRSWT	0	R/W	GTCCRA and GTCCRB Forcible Buffer Operation Writing 1 to this bit forces a buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0 after 1 is written. This bit is read as 0.
b23	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
b25, b24	ADTTA[1:0]	All 0	R/W	GTADTRA Buffer Transfer Timing Select ● Triangle waves b25 b24 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough. ● Saw waves b25 b24 0 0: No transfer Values other than 0 0: Transfer is triggered when the counter underflows (in down-counting), overflows (in up- counting), or is cleared.
b26	ADTDA	0	R/W	GTADTRA Double Buffer Operation 0: Single buffer operation (GTADTBRA ⇔ GTADTRA) 1: Double buffer operation (GTADTDBRA ⇔ GTADTBRA ⇔ GTADTDRA).
b27	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b29, b28	ADTTB[1:0]	All 0	R/W	GTADTRB Buffer Transfer Timing Select ● Triangle waves b29 b28 0 0: No transfer 0 1: Transfer at crest 1 0: Transfer at trough 1 1: Transfer at both crest and trough. ● Saw waves b29 b28 0 0: No transfer Values other than 0 0: Transfer is triggered when the counter underflows (in down-counting), overflows (in up- counting), or is cleared.
b30	ADTDB	0	R/W	GTADTRB Double Buffer Operation 0: Single buffer operation (GTADTBRB ⇔ GTADTRB) 1: Double buffer operation (GTADTDBRB ⇔ GTADTBRB ⇔ GTADTDRB).
b31	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.

GTBER provides settings for the buffer operation and must be set while the GTCNT operation stops.

BD[0] bit (GTCCR Buffer Operation Disable)

The BD[0] bit disables buffer operation using GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, and GTCCRF of the combined GPT.

When GTDTCR.TDE is 1 and when BD[0] is set to 0, GTCCRB does not perform buffer operation and the GTCCRB register is automatically set to a compare match value for a negative-phase waveform with dead time.

BD[1] bit (GTPR Buffer Operation Disable)

The BD[1] bit disables buffer operation using GTPR, GTPBR, and GTPDBR of the combined GPT.

BD[2] bit (GTADTR Buffer Operation Disable)

The BD[2] bit disables buffer operation using GTADTRA, GTADTRB, GTADTBRA, GTADTBRB, GTADTDBRA, and GTADTDBRB of the combined GPT. In event count operation, this bit is not available and the GTADTR buffer operation is not performed.

BD[3] bit (GTDV Buffer Operation Disable)

The BD[3] bit disables buffer operation using GTDVU, GTDVD, GTDBD, and GTDBU of the combined GPT.

When the GTDTCR.TDFER bit is set to 1 and when BD[3] is set to 0, buffer operation is not performed and the GTDVD value is set as a value of GTDVU automatically. In event count operation, this bit is not available and the GTDV buffer operation is not performed.

CCRA[1:0] bits (GTCCRA Buffer Operation)

The CCRA[1:0] bits set buffer operation using GTCCRA, GTCCRC, and GTCCRD of the combined GPT. When buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.*1

CCRB[1:0] bits (GTCCRB Buffer Operation)

The CCRB[1:0] bits set buffer operation using GTCCRB, GTCCRE, and GTCCRF of the combined GPT. When buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.* 1

PR[1:0] bits (GTPR Buffer Operation)

The PR[1:0] bits set buffer operation using GTPR, GTPBR, and GTPDBR of the combined GPT.

CCRSWT bit (GTCCRA and GTCCRB Forcible Buffer Operation)

Writing 1 to the CCRSWT bit forcibly performs a buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0 after 1 is written. This bit is read as 0 and is only valid when counting is stopped with a specified compare match operation.

ADTTA[1:0] bits (GTADTRA Buffer Transfer Timing Select)

The ADTTA[1:0] bits set the transfer timing for buffer operation of GTADTRA, GTADTBRA, and GTADTDBRA. These bits are not available in event count operation.

ADTDA bit (GTADTRA Double Buffer Operation)

The ADTDA bit sets buffer operation with GTADTRA, GTADTBRA, and GTADTDBRA combined. This bit is not available in event count operation.

ADTTB[1:0] bits (GTADTRB Buffer Transfer Timing Select)

The ADTTB[1:0] bits set the transfer timing for buffer operation of GTADTRB, GTADTBRB, and GTADTDBRB. This bit is not available in event count operation.

ADTDB bit (GTADTRB Double Buffer Operation)

The ADTDB bit sets buffer operation using GTADTRB, GTADTBRB, and GTADTDBRB of the combined GPT. This bit is not available in event count operation.

Note 1. The buffer operation mode is fixed in saw-wave one-shot pulse mode, or triangle-wave PWM mode 3 (64-bit transfer at trough).

18.2.18 General PWM Timer Interrupt and A/D Converter Start Request Skipping Setting Register (GTITC)

Address(es): GPT32Em.GTITC H'1004_8044 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	ADTBL	—	ADTAL	—	IVTT[2:0]			IVTC[1:0]		ITLF	ITLE	ITLD	ITLC	ITLB	ITLA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ITLA	0	R/W	GTCCRA Compare Match/Input Capture Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b1	ITLB	0	R/W	GTCCRB Compare Match/Input Capture Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b2	ITLC	0	R/W	GTCCRC Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b3	ITLD	0	R/W	GTCCRD Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b4	ITLE	0	R/W	GTCCRE Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b5	ITLF	0	R/W	GTCCRF Compare Match Interrupt Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b7, b6	IVTC[1:0]	All 0	R/W	OVFn/UNFn Interrupt Skipping Function Select b7 b6 0 0: Do not perform skipping 0 1: Count and skip both overflow and underflow for saw waves and crest for triangle waves 1 0: Count and skip both overflow and underflow for saw waves and trough for triangle waves 1 1: Count and skip both overflow and underflow for saw waves and both crest and trough for triangle waves.

Bit	Bit Name	Initial Value	R/W	Description
b10 to b8	IVTT[2:0]	All 0	R/W	OVFn/UNFn Interrupt Skipping Count Select b10 b8 0 0 0: No skipping 0 0 1: Skipping count of 1 0 1 0: Skipping count of 2 0 1 1: Skipping count of 3 1 0 0: Skipping count of 4 1 0 1: Skipping count of 5 1 1 0: Skipping count of 6 1 1 1: Skipping count of 7.
b11	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b12	ADTAL	0	R/W	GTADTRA A/D Converter Start Request Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b13	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b14	ADTBL	0	R/W	GTADTRB A/D Converter Start Request Link 0: Do not link with OVFn/UNFn interrupt skipping function 1: Link with OVFn/UNFn interrupt skipping function.
b31 to b15	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTITC sets the skipping function for the GTCNT counter overflow (GTPR compare match) interrupt (OVFn) and underflow interrupt (UNFn). It also specifies whether to link other interrupts and A/D converter start requests with the OVFn/UNFn interrupt skipping function. An output disable request to the POEG cannot be linked with the OVFn/UNFn interrupt skipping function. This bit is not available in event count operation.

ITLA bit (GTCCRA Compare Match/Input Capture Interrupt Link)

The ITLA bit specifies whether to link the GTCCRA compare match/input capture interrupt (GTCIA) with the OVFn/UNFn interrupt skipping function.

ITLB bit (GTCCRB Compare Match/Input Capture Interrupt Link)

The ITLB bit specifies whether to link the GTCCRB compare match/input capture interrupt (GTCIB) with the OVFn/UNFn interrupt skipping function.

ITLC bit (GTCCRC Compare Match Interrupt Link)

The ITLC bit specifies whether to link the GTCCRC compare match interrupt (GTCIC) with the OVFn/UNFn interrupt skipping function.

ITLD bit (GTCCRD Compare Match Interrupt Link)

The ITLD bit specifies whether to link the GTCCRD compare match interrupt (GTCID) with the OVFn/UNFn interrupt skipping function.

ITLE bit (GTCCRE Compare Match Interrupt Link)

The ITLE bit specifies whether to link the GTCCRE compare match interrupt (GTCIE) with the OVFn/UNFn interrupt skipping function.

ITLF bit (GTCCRF Compare Match Interrupt Link)

The ITLF bit specifies whether to link the GTCCRF compare match interrupt (GTCIF) with the OVFn/UNFn interrupt skipping function.

IVTC[1:0] bits (OVFn/UNFn Interrupt Skipping Function Select)

The IVTC[1:0] bits set the skipping function for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn).

IVTT[2:0] bits (OVFn/UNFn Interrupt Skipping Count Select)

The IVTT[2:0] bits set the skipping count for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn). When modifying the IVTT[2:0] bits, first set the IVTC[1:0] bits to 00b.

ADTAL bit (GTADTRA A/D Converter Start Request Link)

The ADTAL bit specifies whether to link the GTADTRA A/D converter start request with OVFn/UNFn interrupt skipping function.

ADTBL bit (GTADTRB A/D Converter Start Request Link)

The ADTBL bit specifies whether to link the GTADTRB A/D converter start request with OVFn/UNFn interrupt skipping function.

18.2.19 General PWM Timer Counter (GTCNT)

Address(es): GPT32Em.GTCNT H’1004_8048 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTCNT is a 32-bit read/write counter and can only be written to after counting stops. GTCNT must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited. GTCNT must be set within the range of 0 ≤ GTCNT ≤ GTPR.

18.2.20 General PWM Timer Compare Capture Register n (GTCCRn) (n = A to F)

Address(es): GPT32Em.GTCCRA H’1004_804C + H’0100 × m (m = 0 to 7)
GPT32Em.GTCCRB H’1004_8050 + H’0100 × m (m = 0 to 7)
GPT32Em.GTCCRC H’1004_8054 + H’0100 × m (m = 0 to 7)
GPT32Em.GTCCRE H’1004_8058 + H’0100 × m (m = 0 to 7)
GPT32Em.GTCCRD H’1004_805C + H’0100 × m (m = 0 to 7)
GPT32Em.GTCCRF H’1004_8060 + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTCCRn registers are read/write registers.

GTCCRA and GTCCRB are registers used for both output compare and input capture.

GTCCRC and GTCCRE are compare match registers that can also function as buffer registers for GTCCRA and GTCCRB.

GTCCRD and GTCCRF are compare match registers that can also function as buffer registers for GTCCRC and GTCCRE (double-buffer registers for GTCCRA and GTCCRB).

18.2.21 General PWM Timer Cycle Setting Register (GTPR)

Address(es): GPT32Em.GTPR H'1004_8064 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTPR is a read/write register that sets the maximum count value of GTCNT. For saw waves, the value of (GTPR + 1) is the cycle. For triangle waves, the value of (GTPR value × 2) is the cycle.

18.2.22 General PWM Timer Cycle Setting Buffer Register (GTPBR)

Address(es): GPT32Em.GTPBR H'1004_8068 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTPBR is a read/write register that functions as a buffer register for GTPR.

18.2.23 General PWM Timer Cycle Setting Double-Buffer Register (GTPDBR)

Address(es): GPT32Em.GTPDBR H’1004_806C + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTPDBR is a 32-bit read/write register that functions as a buffer register for GTPBR (double-buffer register for GTPR).

18.2.24 A/D Converter Start Request Timing Register n (GTADTRn) (n = A, B)

Address(es): GPT32Em.GTADTRA H’1004_8070 + H’0100 × m (m = 0 to 7)
GPT32Em.GTADTRB H’1004_807C + H’0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTADTRn registers are 32-bit read/write registers that set the timing of A/D converter start request generation. When the GTADTRn value matches the GTCNT counter value, an A/D converter start request is generated. GTADTRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

18.2.25 A/D Converter Start Request Timing Buffer Register n (GTADTBRn) (n = A, B)

Address(es): GPT32Em.GTADTBRA H'1004_8074 + H'0100 × m (m = 0 to 7)
GPT32Em.GTADTBRB H'1004_8080 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTADTBRn registers are 32-bit read/write registers that function as buffer registers for GTADTRn. GTADTBRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

18.2.26 A/D Converter Start Request Timing Double-Buffer Register n (GTADTDBRn) (n = A, B)

Address(es): GPT32Em.GTADTDBRA H'1004_8078 + H'0100 × m (m = 0 to 7)
GPT32Em.GTADTDBRB H'1004_8084 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTADTDBRn registers are 32-bit read/write registers that function as buffer registers for GTADTBRn (double-buffer registers for GTADTR). GTADTDBRn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

18.2.27 General PWM Timer Dead Time Control Register (GTDTCR)

Address(es): GPT32Em.GTDTCR H'1004_8088 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDFER	—	—	TDBDE	TDBUE	—	—	—	TDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	TDE	0	R/W	Negative-Phase Waveform Setting 0: Set GTCCRB without using GTDVU and GTDVD. 1: Use GTDVU and GTDVD to set the compare match value for negative-phase waveform with automatic dead time in GTCCRB.
b3 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b4	TDBUE	0	R/W	GTDVU Buffer Operation Enable 0: Disable GTDVU buffer operation 1: Enable GTDVU buffer operation.
b5	TDBDE	0	R/W	GTDVD Buffer Operation Enable 0: Disable GTDVD buffer operation 1: Enable GTDVD buffer operation.
b7, b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b8	TDFER	0	R/W	GTDVD Setting 0: Set GTDVU and GTDVD separately 1: Automatically set the value written to GTDVU to GTDVD.
b31 to b9	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTDTCR enables automatic setting of a compare match value for negative-phase waveform with dead time.

TDE bit (Negative-Phase Waveform Setting)

The TDE bit specifies whether to use GTDVU and GTDVD. When GTDVU and GTDVD are used, the compare match value for a negative-phase waveform with dead time obtained by the compare match value of a positive-phase waveform (GTCCRA) and the dead time value (GTDVU and GTDVD), is automatically set in GTCCRB.

The TDE bit setting is ignored in saw-wave PWM mode, and automatic setting does not take place.

The GTCCRB value is automatically set and has the following upper and lower limit values. If the obtained GTCCRB value is not within the upper or lower limit, the following limit value is set in GTCCRB and the GTST.DTEF flag is set to 1. However, in triangle waves, when the obtained GTCCRB value exceeds the upper limit value, the GTST.DTEF flag is set to 0.

- Triangle waves
  Upper limit value: GTPR – 1
  Lower limit value: 1 in up-counting, 0 in down-counting
- Saw-wave one-shot pulse mode
  Upper limit value: GTPR
  Lower limit value: 0

TDBUE bit (GTDVU Buffer Operation Enable)

The TDBUE bit enables buffer operation with GTDVU and GTDBU combined. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves.

TDBDE bit (GTDVD Buffer Operation Enable)

The TDBDE bit enables buffer operation with GTDVD and GTDBD combined. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves. When this bit and the TDFER bit are set to 1 simultaneously, the TDFER bit setting is given priority.

TDFER bit (GTDVD Setting)

The TDFER bits sets whether or not the value written to GTDVU is also set to GTDVD automatically.

18.2.28 General PWM Timer Dead Time Value Register n (GTDVn) (n = U, D)

Address(es): GPT32Em.GTDVU H'1004_808C + H'0100 x m (m = 0 to 7)
GPT32Em.GTDVD H'1004_8090 + H'0100 x m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTDVn is a 32-bit read/write register that sets the dead time for generating PWM waveforms with dead time. GTDVU is used for up-counting and GTDVD is used for down-counting.

It is prohibited to set the GTDVn register with a value larger than or equal to that of the GTPR register.

Do not set the change point of waveform beyond the count cycle period, when the automatic dead time setting function is used. By reading the GTCCRB register, can be seen the change point for the reverse-phase waveform after the dead time added, which is set in the automatic dead time setting function.

When GTDVn is used, writing to GTCCRB is not allowed. When this register is set to 0, waveforms without dead time are output. GTDVn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited.

When GTDVn buffer operation is enabled, GTDBn can be written anytime. The value of GTDBn is transferred to GTDVn at the end of the count cycle period. When the GTDVn buffer operation is disabled, set the CST bit in the GTCR register to stop the GPT with before changing GTDVn to a new value. When GTDVn buffer operation is disabled, to change GTDVn to a new value, stop the GPT with the CST bit in the GTCR register.

18.2.29 General PWM Timer Dead Time Buffer Register n (GTDBn) (n = U, D)

Address(es): GPT32Em.GTDBU H'1004_8094 + H'0100 × m (m = 0 to 7)
GPT32Em.GTDBD H'1004_8098 + H'0100 × m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

GTDBn is a 32-bit read/write register that functions as a buffer register for GTDVn.

18.2.30 General PWM Timer Output Protection Function Status Register (GTSOS)

Address(es): GPT32Em.GTSOS H’1004_809C + H’0100 x m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	SOS[1:0]	All 0	R	Output Protection Function Status b1 b0 0 0: Normal operation 0 1: Protected state (set GTCCRA = 0 during transfer at trough or crest) 1 0: Protected state (set GTCCRA ≥ GTPR during transfer at trough) 1 1: Protected state (set GTCCRA ≥ GTPR during transfer at crest).
b7 to b2	—	All 0	R	Reserved These bits are read as 0.
b9, b8	—	All 0	R	Reserved The read value is undefined.
b31 to b10	—	All 0	R	Reserved These bits are read as 0.

GTSOS is a status register that indicates the status of the output protection function. The output protection function is enabled only when the dead time is automatically set (GTDTCR.TDE bit = 1) in triangle-wave mode.

SOS[1:0] bits (Output Protection Function Status)

The SOS[1:0] bits indicate the status of the output protection function in triangle-wave PWM mode.

18.2.31 General PWM Timer Output Protection Function Temporary Release Register (GTSOTR)

Address(es): GPT32Em.GTSOTR H’1004_80A0 + H’0100 x m (m = 0 to 7)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOTR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	SOTR	0	R/W	Output Protection Function Temporary Release 0: Do not release protected state 1: Release protected state.
b31 to b1	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

GTSOTR temporarily releases the protected state of GTIOCB pin output when output protection is set. The protected state can be released only when GTSOS.SOS[1:0] bits = 10b (protected state in which GTCCRA ≥ GTPR has occurred during transfer at trough). The protected state cannot be released in any other case.

SOTR bit (Output Protection Function Temporary Release)

The SOTR bit specifies whether to temporarily release the protected state of the GTIOCB pin output in an output protected state. When the SOTR bit is set to 1, the output protection function is canceled from the first trough. When the SOTR bit is set to 0, output protection resumes from the first trough.

18.3 Operation

18.3.1 Basic Operation

Each channel has a 32-bit timer that performs a periodic count operation using the count clock and hardware sources. The count function provides both up-counting and down-counting. The GTPR controls the count cycle. When the GTCNT counter value matches the value in GTCCRA or GTCCRB, the output from the associated pin GTIOCA or GTIOCB can be changed. GTCCRA or GTCCRB can be used as an input capture register with hardware resources. GTCCRC and GTCCRD can function as buffer registers for GTCCRA. GTCCRE and GTCCRF can function as buffer registers for GTCCRB.

18.3.1.1 Counter Operation

(1) Counter start and stop

The counter of each channel starts the count operation by setting GTCR.CST to 1. The GTCR.CST bit value is changed by following sources.

- Writing to GTCR register
- Writing 1 to the bit in GTSTR associated with the GPT channel number when the GTSSR.CSTRT bit set to 1
- Writing 1 to the bit in GTSTP associated with the GPT channel number when the GTPSR.CSTOP bit set to 1
- The hardware source selected in the GTSSR register
- The hardware source selected in the GTPSR register.

(2) Periodic count operation in up-counting by count clock

The GTCNT counter in each channel starts up-counting when the associated GTCR.CST bit is set to 1 with GTUPSR and GTDNSR registers set to H’00000000. When the GTCNT value changes from the GTPR value to 0 (overflow), the GTST.TCFPO flag is set to 1. When GTCNT overflows, up-counting resumes from H’00000000.

Figure 18.2 shows an example of a periodic count operation in up-counting.

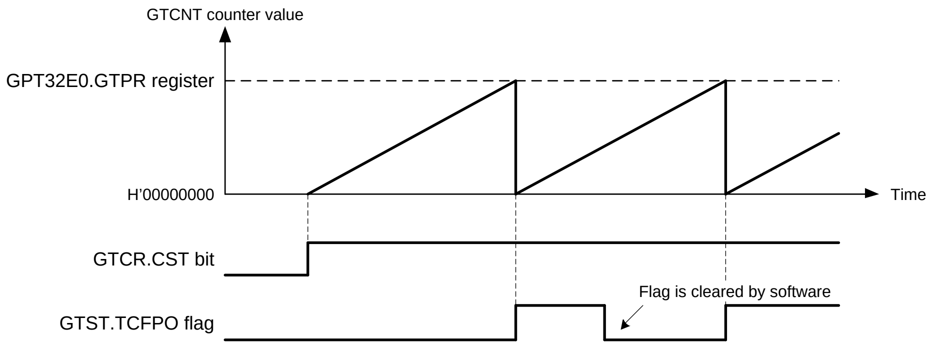
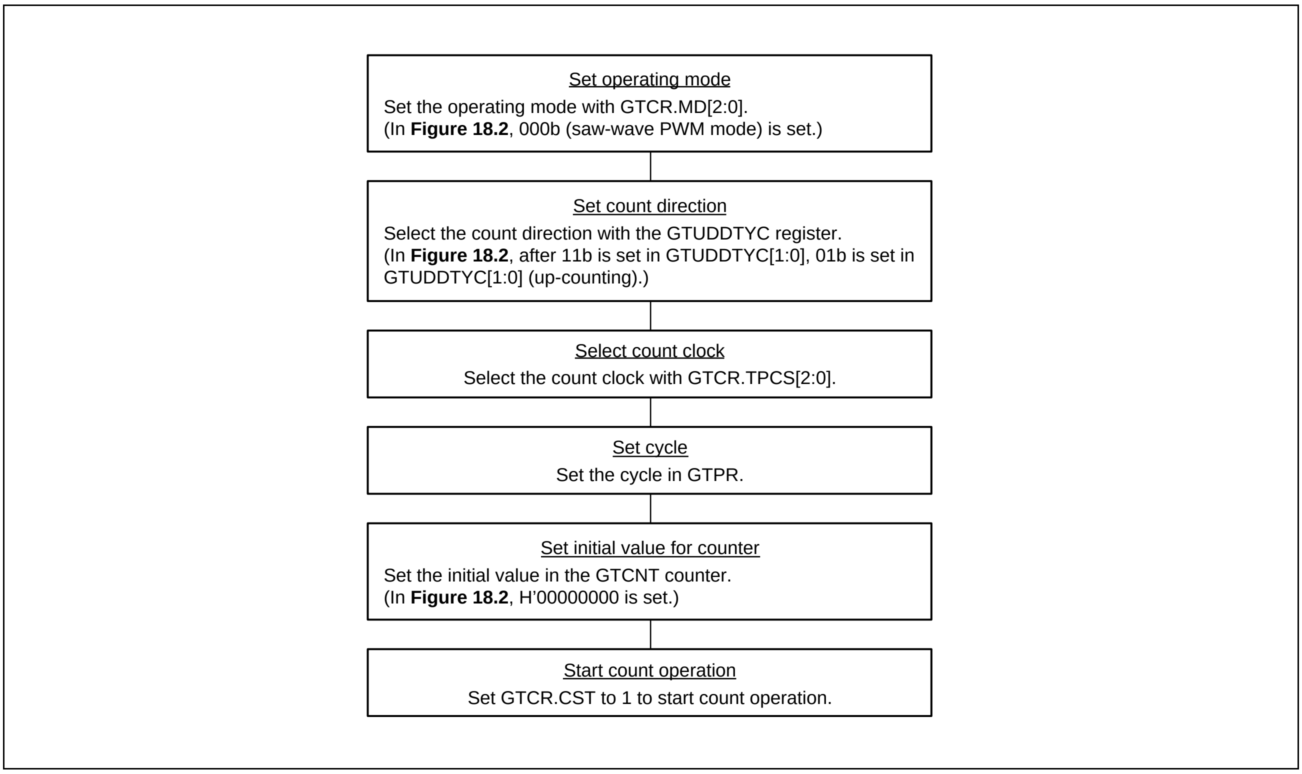


Figure 18.2 Example of periodic count operation in up-counting by the count clock

Figure 18.3 shows an example setting for periodic count operation in up-counting.



Set operating mode
Set the operating mode with GTCR.MD[2:0].
(In **Figure 18.2**, 000b (saw-wave PWM mode) is set.)

Set count direction
Select the count direction with the GTUDDTYC register.
(In **Figure 18.2**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.
(In **Figure 18.2**, H'00000000 is set.)

Start count operation
Set GTCR.CST to 1 to start count operation.

Figure 18.3 Example setting for a periodic count operation in up-counting by the count clock

(3) Periodic count operation in down-counting by count clock

The GTCNT counter in each channel can perform down-counting by setting GTUDDTYC.UD with GTUPSR and GTDNSR registers set to H’00000000. When GTCNT changes from 0 to the GTPR value (underflow), GTST.TCFPU is set to 1. When the GTCNT counter underflows, down-counting resumes from the GTPR value.

Figure 18.4 shows an example of periodic count operation in down-counting by the count clock.

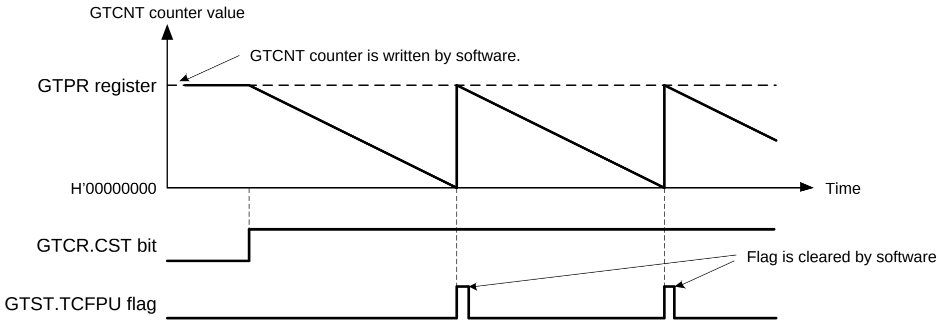
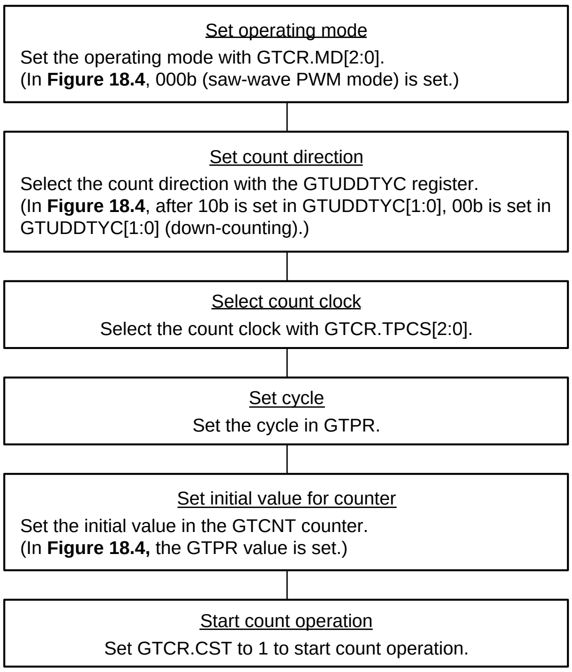


Figure 18.4 Example of periodic count operation in down-counting by the count clock

Figure 18.5 shows an example setting for periodic count operation in down-counting by the count clock.



Set operating mode
Set the operating mode with GTCR.MD[2:0].
(In **Figure 18.4**, 000b (saw-wave PWM mode) is set.)

Set count direction
Select the count direction with the GTUDDTYC register.
(In **Figure 18.4**, after 10b is set in GTUDDTYC[1:0], 00b is set in GTUDDTYC[1:0] (down-counting).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.
(In **Figure 18.4,** the GTPR value is set.)

Start count operation
Set GTCR.CST to 1 to start count operation.

Figure 18.5 Example setting for periodic count operation in down-counting by count clock

(4) Event count operation in up-counting using hardware sources

The GTCNT counter in each channel can perform up-counting using hardware sources as set in GTUPSR.

When GTUPSR is set to enable, the count clock selected in GTCR.TPCS[2:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time, the GTCNT counter value does not change. The overflow behavior for up-counting using hardware sources is the same as for up-counting by the count clock.

When GTCR.CST bit is set to 1 to count up using hardware sources, the count operation is enabled. After GTCR.CST is set to 1, the counter cannot count up for 1 clock cycle as specified in GTCR.TPCS[2:0] because the count operation is synchronized by the count clock selected by GTCR.TPCS[2:0]. Set the GTCR.TPCS[2:0] bits to 000b to select counting up of single cycles of the P0ϕ clock with a delay of one clock cycle after the GTCR.CST bit has been set to 1.

Figure 18.6 shows an example of a periodic count operation in up-counting by a hardware resource (rising edge of GTETRGA pin).

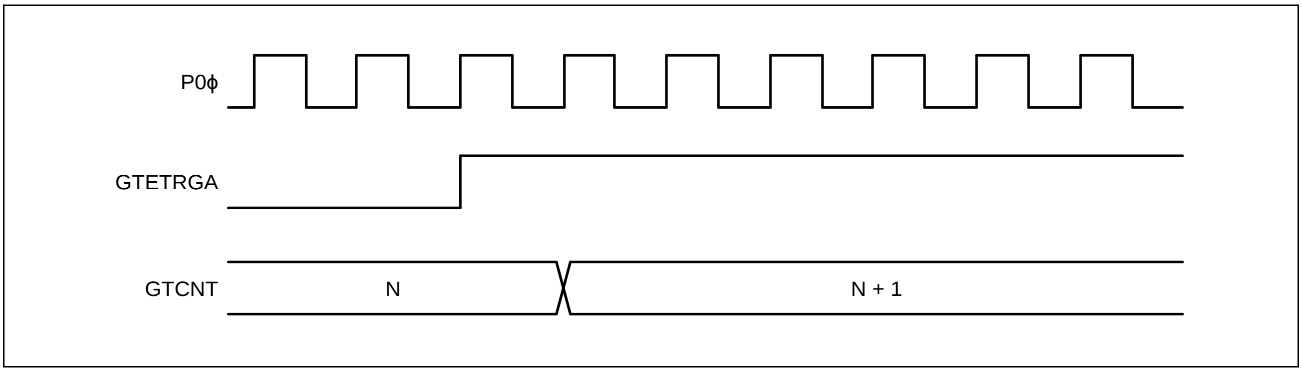
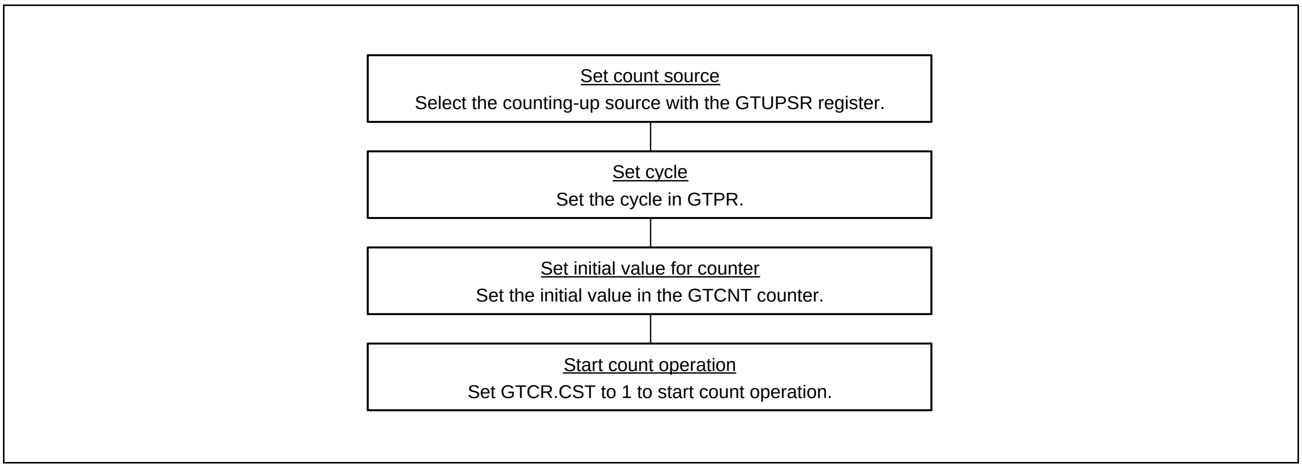


Figure 18.6 Example of periodic count operation in up-counting using hardware sources

Figure 18.7 shows an example setting for periodic count operation in up-counting by the count clock.


Set count source
Select the counting-up source with the GTUPSR register.

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.

Start count operation
Set GTCR.CST to 1 to start count operation.

Figure 18.7 Example setting for an event count operation in up-counting using hardware sources

(5) Event count operation in down-counting using hardware sources

The GTCNT counter in each channel can perform down-counting using hardware sources set in the GTDNSR. When GTDNSR is set to enable, the count clock selected in GTCR.TPCS[2:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time,

GTCNT counter value does not change. The underflow behavior for down-counting using hardware sources is the same as for down-counting by the count clock.

When GTCR.CST bit is set to 1 to count down using hardware sources, the count operation is enabled. After GTCR.CST is set to 1, the counter cannot count down for 1 clock cycle as specified in GTCR.TPCS[2:0] because the count operation is synchronized with the count clock selected by GTCR.TPCS[2:0]. Set the GTCR.TPCS[2:0] bits to 000b to select counting up of single cycles of the P0ϕ clock with a delay of one clock cycle after the GTCR.CST bit has been set to 1.

Figure 18.8 shows an example of a periodic count operation in down-counting by a hardware resource (rising edge of GTETRGA pin).

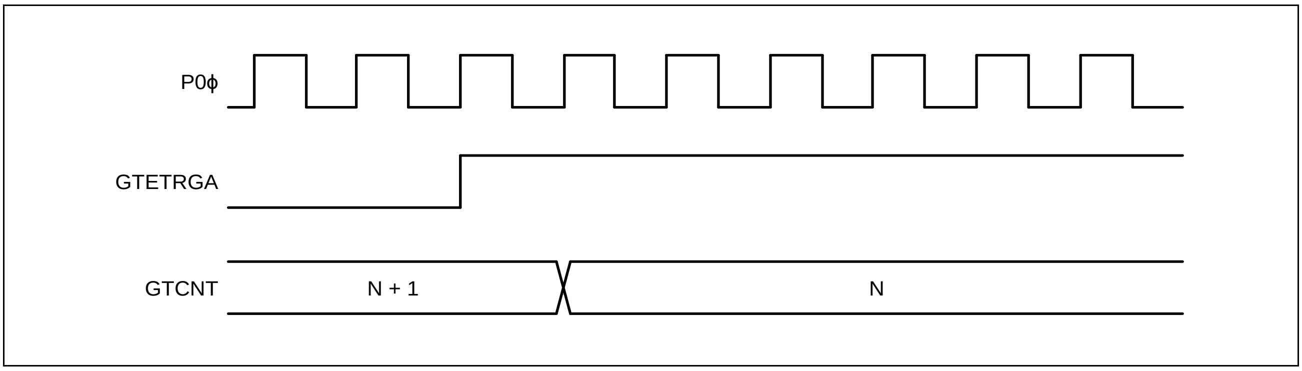


Figure 18.8 Example of event count operation in down-counting using hardware sources

Figure 18.9 shows an example setting for a periodic count operation in down-counting using a hardware resource.

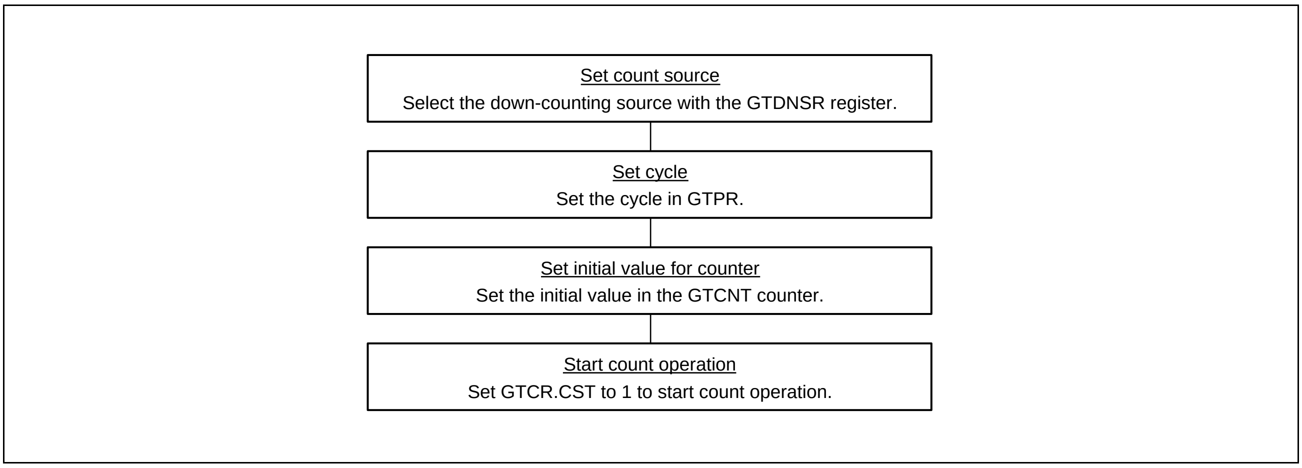


Figure 18.9 Example setting for an event count operation in down-counting using hardware sources

(6) Counter clear operation

The counter of each channel is cleared by either of the following sources.

- Writing 0 to GTCNT register
- Writing 1 to the bit in GTCLR associated with the GPT channel number when the GTCSR.CCLR bit set to 1
- The hardware source selected in GTCSR register.

Writing to the GTCNT register is prohibited during count operation. The GTCNT counter can be cleared both by writing 1 to the GTCLR and by the clear request of hardware sources, whether GTCNT is counting (GTCR.CST = 1) or not (GTCR.CST = 0).

For saw waves selected by setting GTCR.MD[2:0] and the count direction flag showing down-counting (GTST.TUCF = 0), the GTCNT register is set to the value of the GTPR register when the counter is cleared by writing 1 to the GTCLR register or by the hardware source specified in the GTCSR register. When not in saw waves mode and down-counting, the GTCNT register is set to 0 when writing 1 to the GTCLR register and when clearing by hardware sources are performed.

In event count operation when at least 1 bit in GTUPSR or GTDNSR is set to 1, after clear sources occur, both writing to GTCLR register and clearing by hardware sources are performed immediately to synchronize with P0ϕ. If other settings are used, clear is synchronized with the counter clock selected by GTCR.TPCS[2:0].

18.3.1.2 Waveform Output by Compare Match

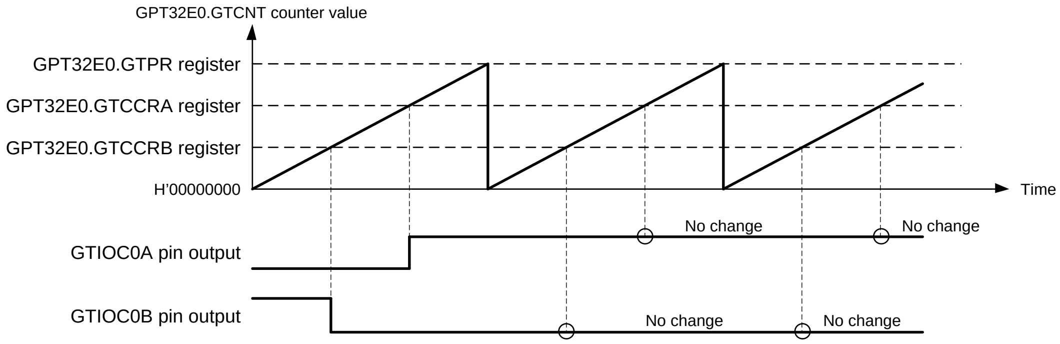
Compare match means that the GTCNT counter value matches the value of GTCCRA or GTCCRB. When a compare match occurs, the compare match flag is generated synchronously with the count clock including the event count. At the same time the GPT can output low, high, or toggle output from the associated GTIOCA or GTIOCB output pin. In addition, the GTIOCA or GTIOCB pin output can be low, high, or toggle at the cycle end which is determined by GTPR. The cycle end is:

- For saw waves in up-counting – when GTCNT changes from the GTPR value to 0 (overflow)
- For saw waves in down-counting – when GTCNT changes from 0 to GTPR value (underflow)
- For saw waves – when the GTCNT counter is cleared
- For triangle waves – when the GTCNT changes from 0 to 1 (trough).

(1) Low and high output

Figure 18.10 shows an example of low and high output operation by a compare match of GTCCRA and GTCCRB.

In this example, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that high is output from the GTIOC0A pin by a GPT32E0.GTCCRA compare match, and low is output from the GTOC0B pin by a GPT32E0.GTCCRB compare match. The pin level does not change when the specified level and pin level match.



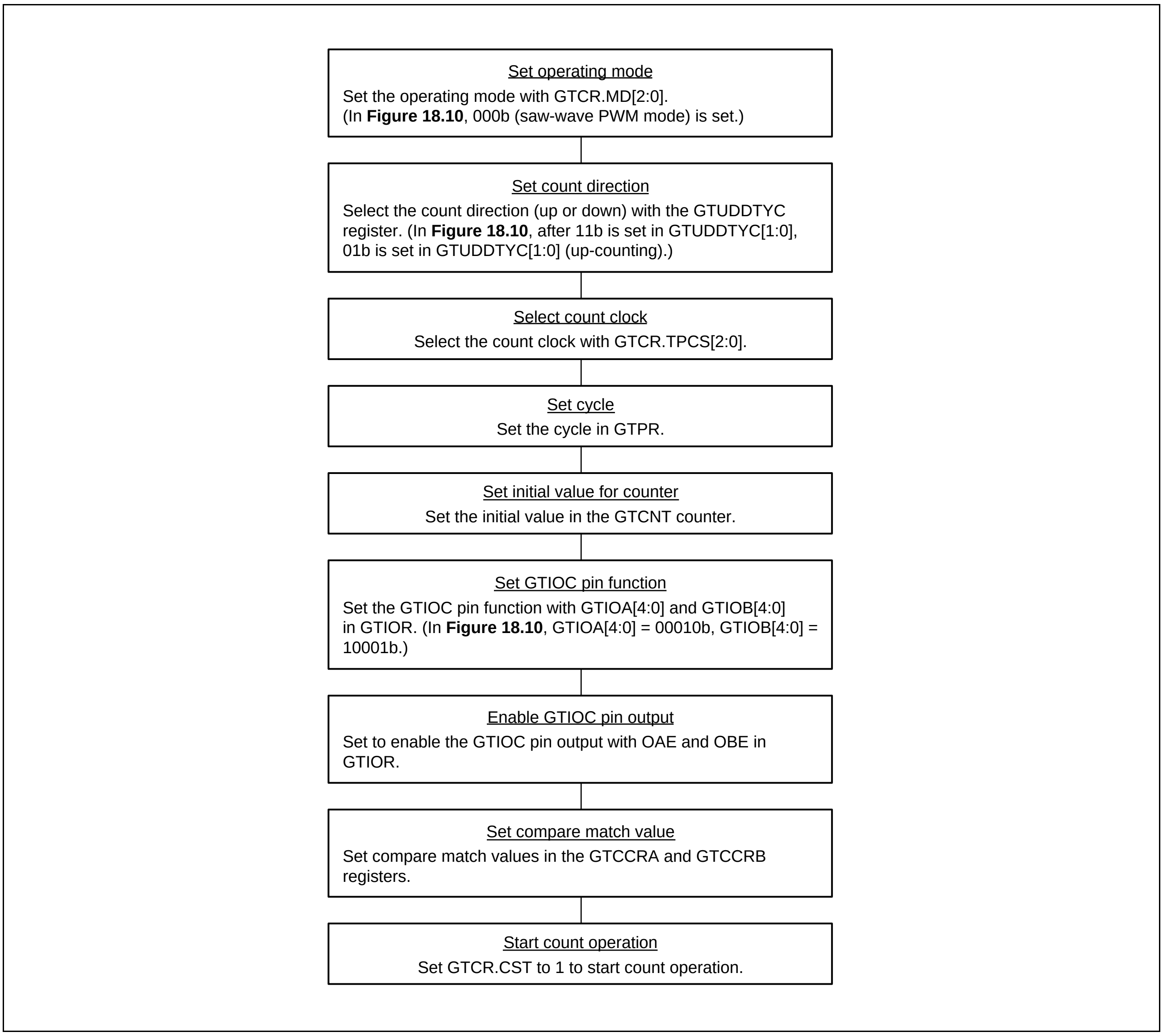
[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits: Initial output is low, high output at compare match, output retained at cycle end

GPT32E0.GTIOR.GTIOB[4:0] bits: Initial output is high, low output at compare match, output retained at cycle end

Figure 18.10 Example of low and high output operation

Figure 18.11 shows an example setting for low output and high output operation.



Set operating mode

Set the operating mode with GTCR.MD[2:0].
(In **Figure 18.10**, 000b (saw-wave PWM mode) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.10**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set GTIOC pin function

Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR. (In **Figure 18.10**, GTIOA[4:0] = 00010b, GTIOB[4:0] = 10001b.)

Enable GTIOC pin output

Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set compare match value

Set compare match values in the GTCCRA and GTCCRB registers.

Start count operation

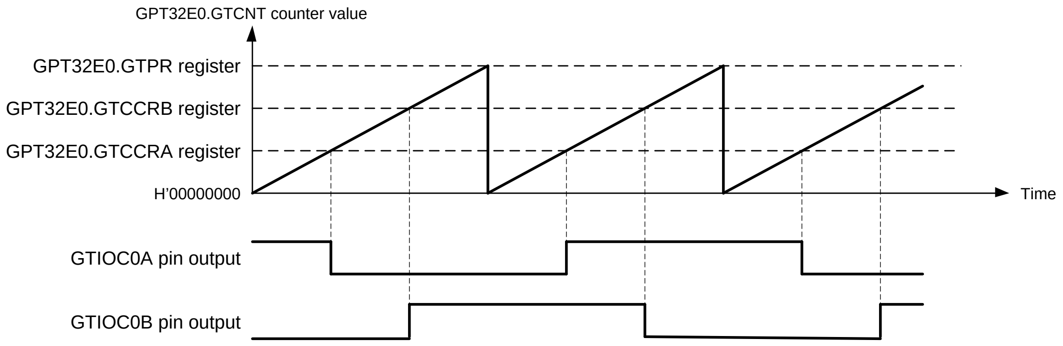
Set GTCR.CST to 1 to start count operation.

Figure 18.11 Example setting for low output and high output operation

(2) Toggled output

Figure 18.12 and **Figure 18.13** show examples of toggled output operation by compare matches of GTCCRA and GTCCRB. In **Figure 18.12**, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that the GTIOC0A pin output by a GPT32E0.GTCCRA compare match and GTIOC0B pin output by a GPT32E0.GTCCRB compare match are toggled.

Figure 18.13, the GPT32E0.GTCNT counter performs up-counting, and settings are made so that the GTIOC0A output is toggled by a compare match of GPT32E0.GTCCRA and the GTIOC0B output is toggled at the cycle end.

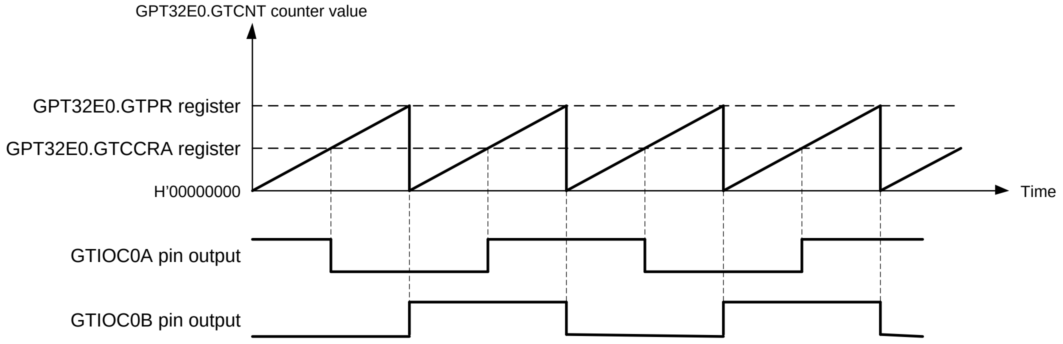


[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits: Initial output is high, output toggled at compare match, output retained at cycle end

GPT32E0.GTIOR.GTIOB[4:0] bits: Initial output is low, output toggled at compare match, output retained at cycle end

Figure 18.12 Example of toggled output operation (1)



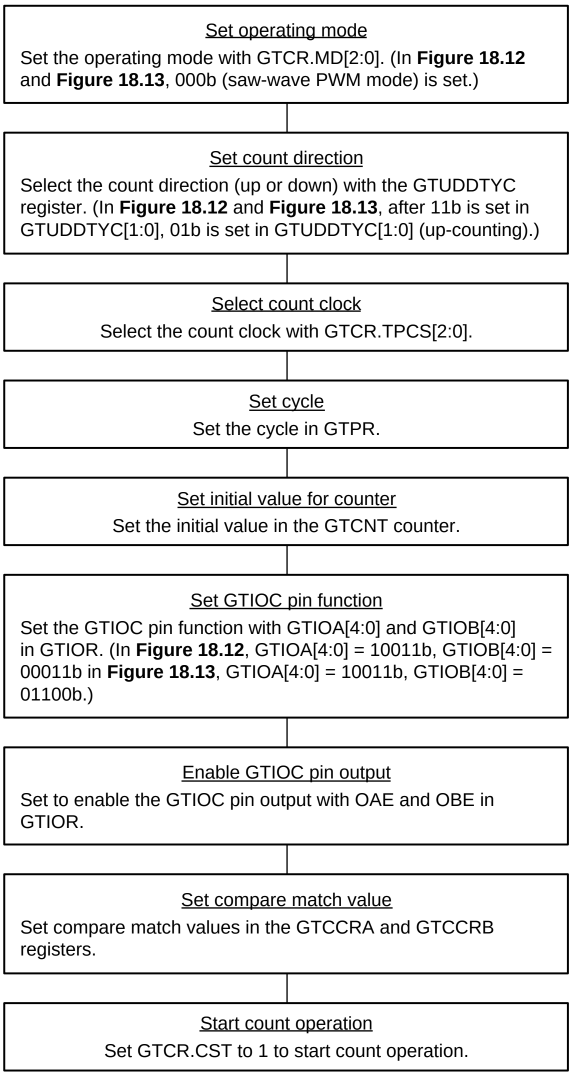
[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits: Initial output is high, output toggled at compare match, output retained at cycle end

GPT32E0.GTIOR.GTIOB[4:0] bits: Initial output is low, output retained at compare match, output toggled at cycle end

Figure 18.13 Example of toggled output operation (2)

Figure 18.14 shows an example setting for toggled output operation.



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.12** and **Figure 18.13**, 000b (saw-wave PWM mode) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.12** and **Figure 18.13**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set GTIOC pin function

Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR. (In **Figure 18.12**, GTIOA[4:0] = 10011b, GTIOB[4:0] = 00011b in **Figure 18.13**, GTIOA[4:0] = 10011b, GTIOB[4:0] = 01100b.)

Enable GTIOC pin output

Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set compare match value

Set compare match values in the GTCCRA and GTCCRB registers.

Start count operation

Set GTCR.CST to 1 to start count operation.

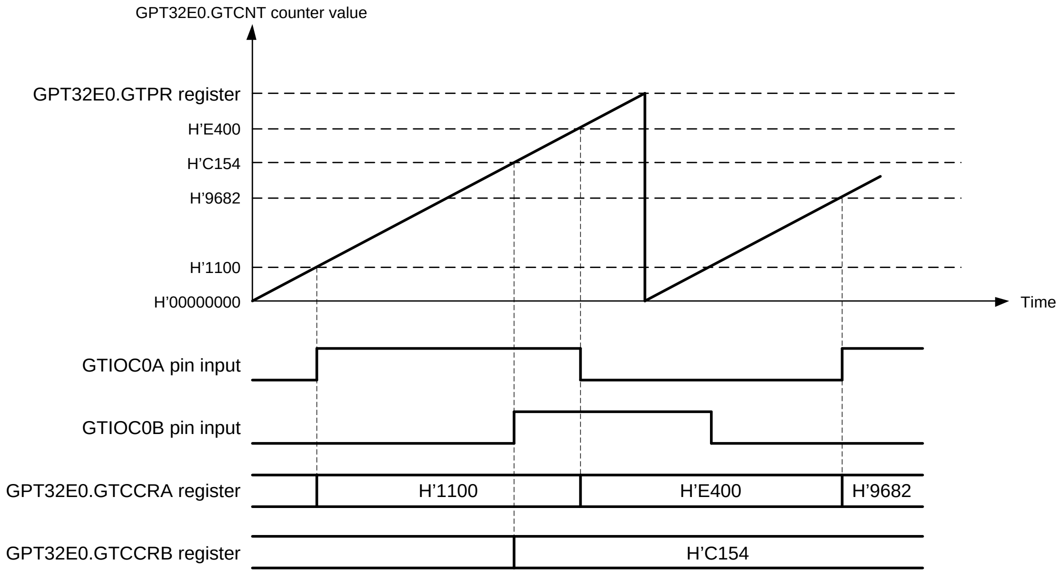
Figure 18.14 Example setting for toggled output operation

18.3.1.3 Input Capture Function

The GTCNT counter value can be transferred to either GTCCRA or GTCCRB on detection of the hardware source that is set in GTICASR and GTICBSR.

Figure 18.15 shows an example of the input capture function.

In this example, the GPT32E0.GTCNT counter performs up-counting by the count clock, and settings are made so that an input capture is performed to GTICCRA at both edges of the GTIOC0A input pin and to GTICCRB on the rising edge of the GTIOC0B input pin.



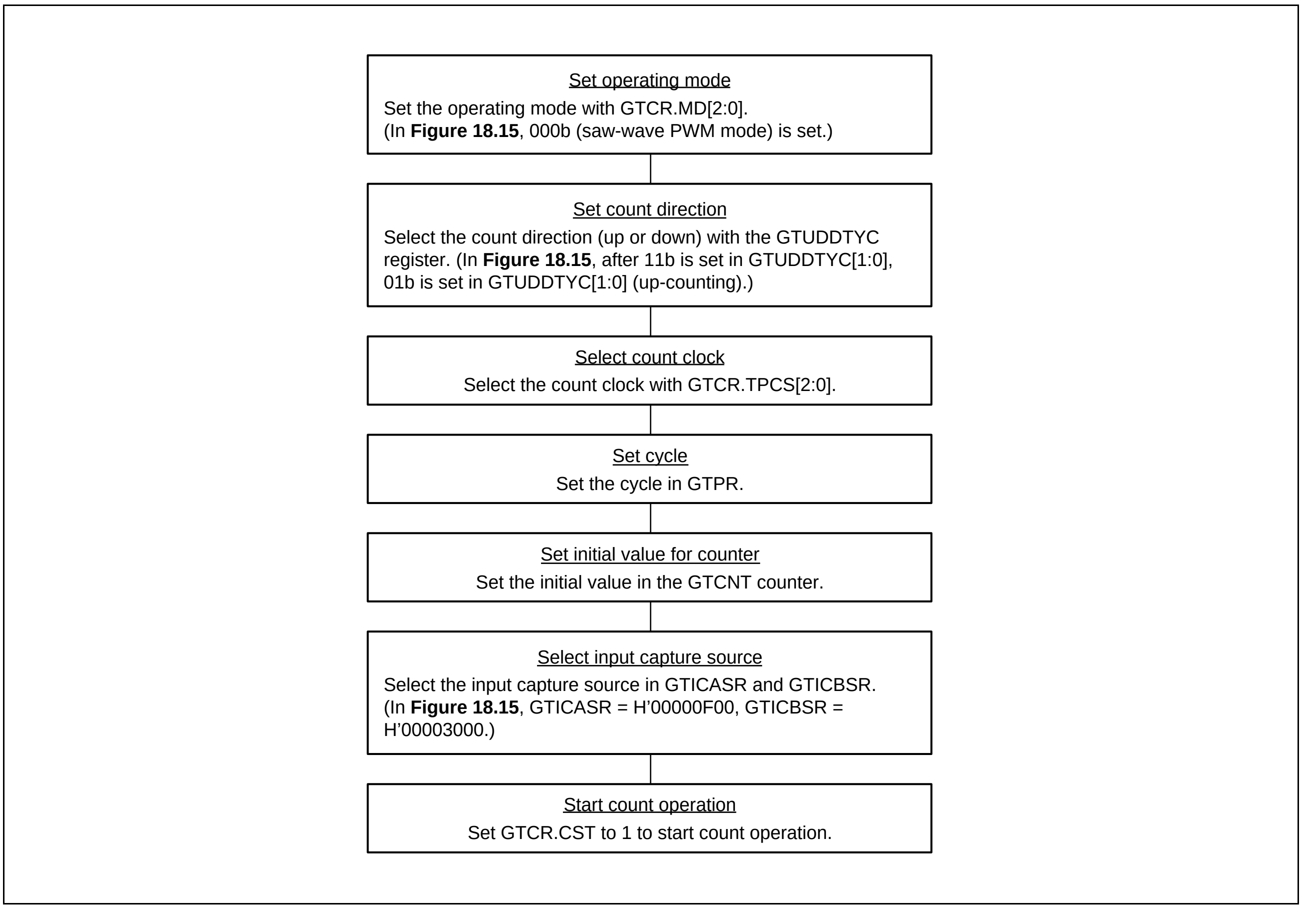
[Setting examples]

GTICASR setting input capture at both edges

GTICBSR setting input capture at the rising edge

Figure 18.15 Example of input capture operation

Figure 18.16 shows an example setting for an input capture operation with count operation by the count clock.



Set operating mode
Set the operating mode with GTCR.MD[2:0].
(In **Figure 18.15**, 000b (saw-wave PWM mode) is set.)

Set count direction
Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.15**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.

Select input capture source
Select the input capture source in GTICASR and GTICBSR.
(In **Figure 18.15**, GTICASR = H'00000F00, GTICBSR = H'00003000.)

Start count operation
Set GTCR.CST to 1 to start count operation.

Figure 18.16 Example setting for input capture operation

18.3.2 Buffer Operation

The following buffer operations can be set with GTBER:

- GTPR, GTPBR, and GTPDBR
- GTCCRA, GTCCRC, and GTCCRD
- GTCCRB, GTCCRE, and GTCCRF
- GTADTRA, GTADTBRA, and GTADTDBRA
- GTADTRB, GTADTBRB, and GTADTDBRB.

The following buffer operations can be set with GTDTCR:

- GTDVU and GTDBU
- GTDVD and GTDBD.

18.3.2.1 GTPR Register Buffer Operation

GTPBR can function as a buffer register for GTPR, and GTPDBR can function as a buffer register for GTPBR (double-buffer register for GTPR). The buffer transfer is performed at an overflow during up-counting or an underflow during down-counting in saw-wave mode or in event count, and at a trough in triangle-wave mode.

In saw-wave mode or in event count, the buffer transfer is performed when the following counter clear operations occur during counting:

- Clear by hardware sources (the clear source is selected in GTCSR[23:0])
- Clear by software (when GTCSR.CCLR bit is 1 and GTCLR[n] bit is set to 1, n = channel number).

To set GTPR to function as double buffer, set GTBER.PR[1:0] to 10b or 11b. To set GTPR to not function as a buffer, set GTBER.PR[1:0] to 00b.

Figure 18.17 to **Figure 18.19** show examples of GTPR buffer operation and **Figure 18.20** shows an example setting for GTPR buffer operation.

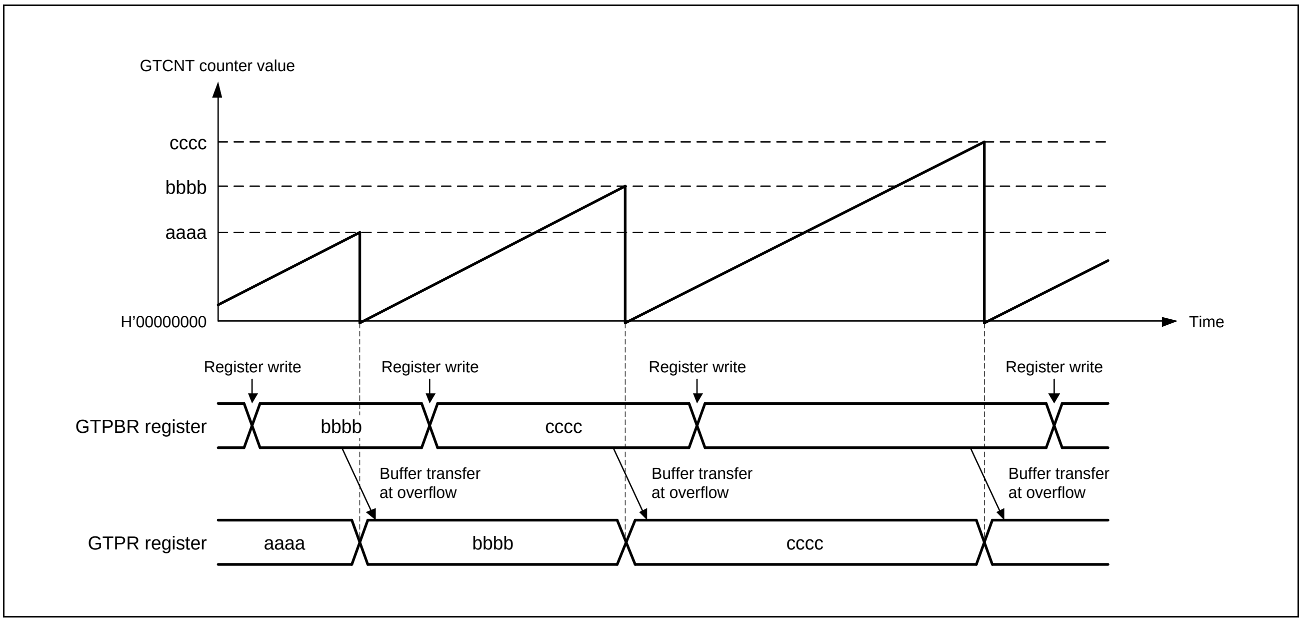


Figure 18.17 Example of GTPR buffer operation with saw waves in up-counting

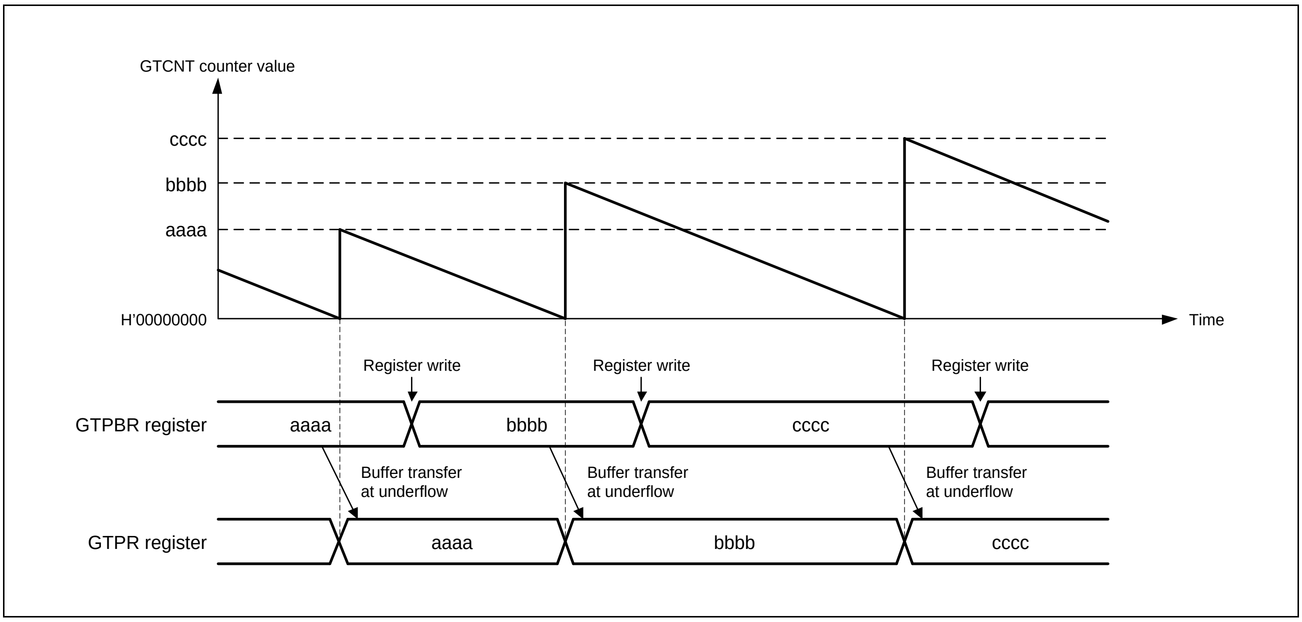


Figure 18.18 Example of GTPR buffer operation with saw waves in down-counting

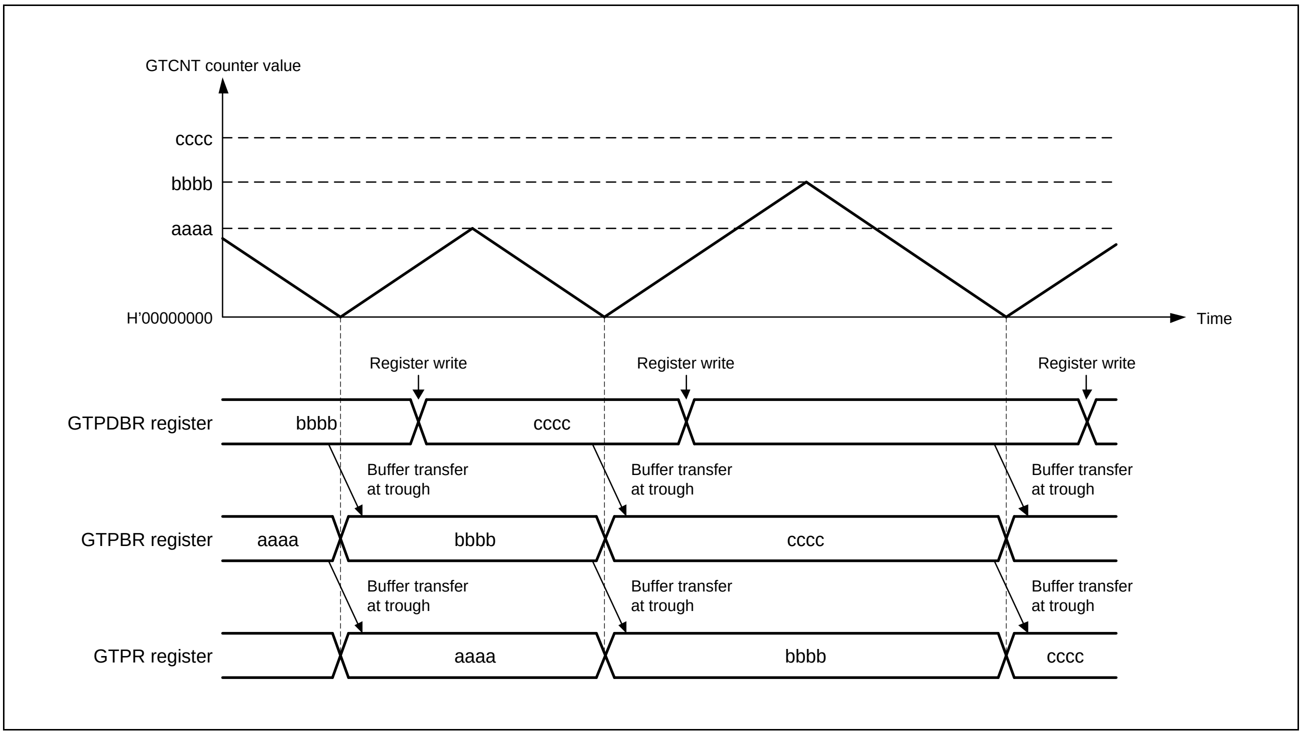
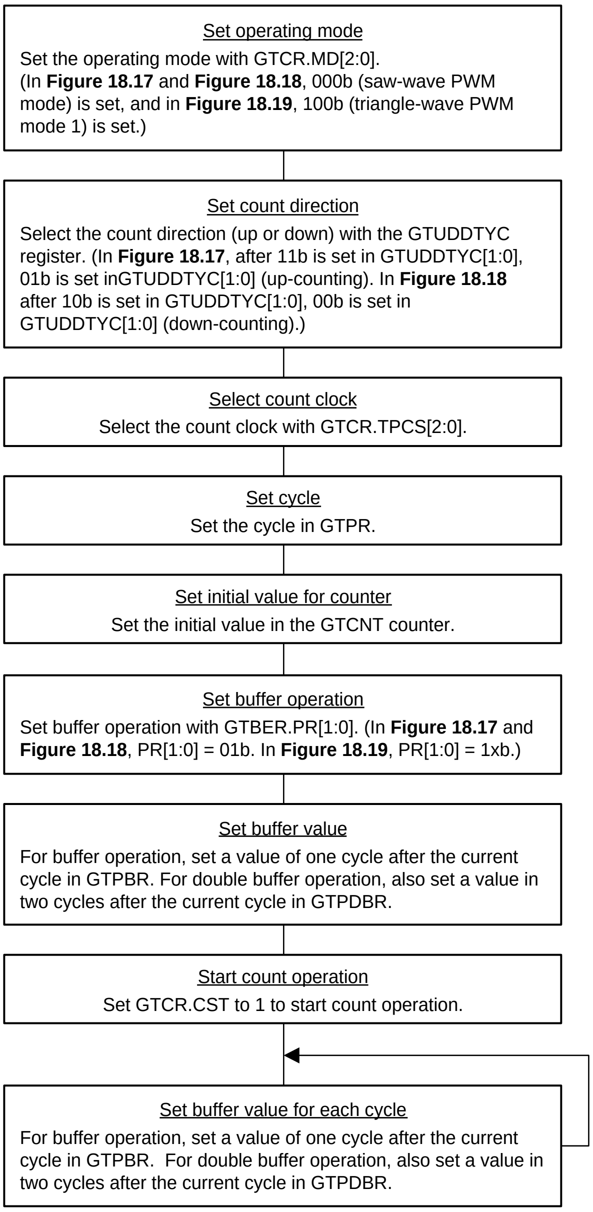


Figure 18.19 Example of GTPR double buffer operation with triangle waves



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.17** and **Figure 18.18**, 000b (saw-wave PWM mode) is set, and in **Figure 18.19**, 100b (triangle-wave PWM mode 1) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.17**, after 11b is set in GTUDDTYC[1:0], 01b is set inGTUDDTYC[1:0] (up-counting). In **Figure 18.18** after 10b is set in GTUDDTYC[1:0], 00b is set in GTUDDTYC[1:0] (down-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set buffer operation

Set buffer operation with GTBER.PR[1:0]. (In **Figure 18.17** and **Figure 18.18**, PR[1:0] = 01b. In **Figure 18.19**, PR[1:0] = 1xb.)

Set buffer value

For buffer operation, set a value of one cycle after the current cycle in GTPBR. For double buffer operation, also set a value in two cycles after the current cycle in GTPDBR.

Start count operation

Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle

For buffer operation, set a value of one cycle after the current cycle in GTPBR. For double buffer operation, also set a value in two cycles after the current cycle in GTPDBR.

Figure 18.20 Example setting for GTPR buffer operation

18.3.2.2 Buffer Operation for GTCCRA and GTCCRB

GTCCRC can function as the GTCCRA buffer register and GTCCRD can function as the GTCCRC buffer register (double-buffer register for GTCCRA). Similarly, GTCCRE can function as the GTCCRB buffer register and GTCCRF can function as the GTCCRE buffer register (double-buffer register for GTCCRB).

To set GTCCRA or GTCCRB to function as a double buffer, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 10b or 11b. For single buffer operation, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 01b. To set GTCCRA or GTCCRB to not function as a buffer, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 00b.

(1) When GTCCRA or GTCCRB functions as an output compare register

Buffer transfer occurs in the following situations:

- Buffer transfer by overflow or underflow
  Buffer transfer is performed at an overflow (during up-counting) or an underflow (during down-counting) in saw-wave mode or in event count operation. In triangle-wave mode, buffer transfer is performed at a trough (triangle-wave PWM mode 1) or a crest and trough (triangle-wave PWM mode 2).
- Buffer transfer by counter clear
  In saw-wave mode or in event count operation, during counting, buffer transfer (which is the same as an overflow during up-counting or an underflow during down-counting) is performed by the counter clear sources the same as shown in **Section 18.3.2.1, GTPR Register Buffer Operation**. In triangle-wave mode, buffer transfer is not performed by the counter clear.
- Forcible buffer transfer
  When 1 is written to the GTBER.CCRSWT bit while counting is stopped, the GTCCRA and the GTCCRB register buffer transfer are performed forcibly in saw-wave mode, in event count operation and in triangle-wave mode. Additionally, buffer transfer from the GTCCRD register to temporary register A and from the GTCCRF register to temporary register B are performed in saw-wave 1 shot pulse mode or triangle-wave PWM mode 3.

Figure 18.21 to **Figure 18.23** show examples of GTCCRA and GTCCRB buffer operation and **Figure 18.24** shows an example setting for GTCCRA and GTCCRB buffer operation.

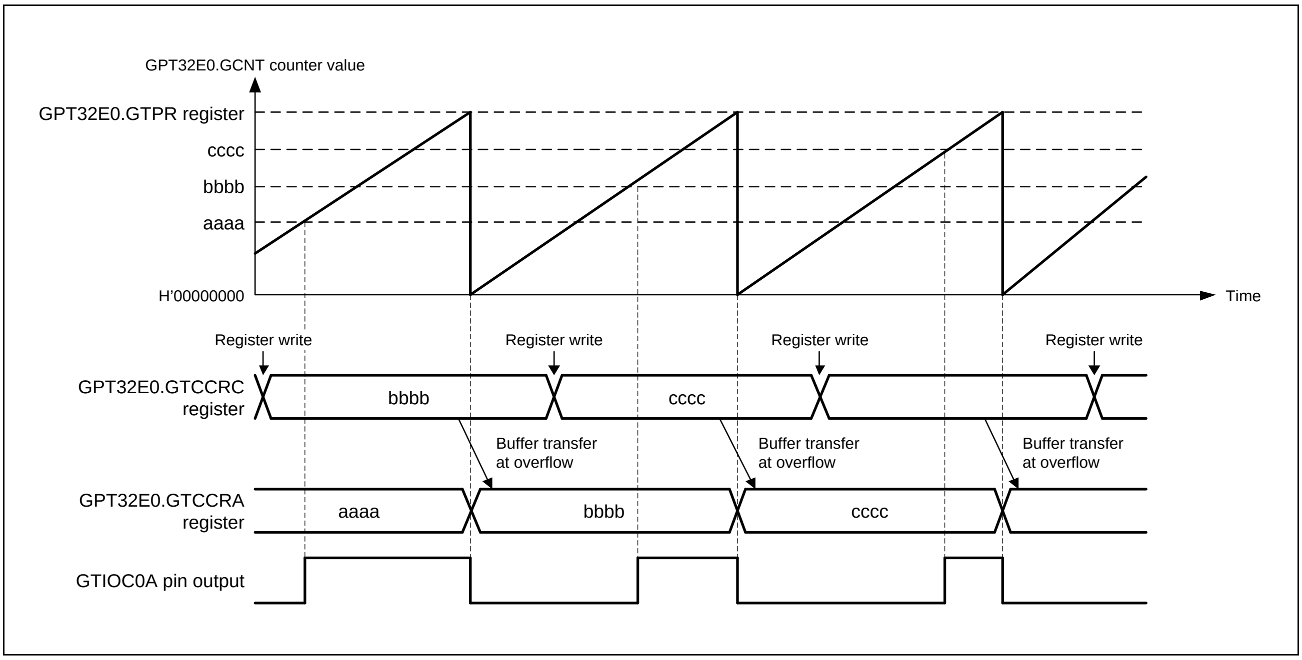


Figure 18.21 Example of GTCCRA and GTCCRB buffer operation with output compare, saw waves in up-counting, high output at GTCCRA compare match, and low output at cycle end

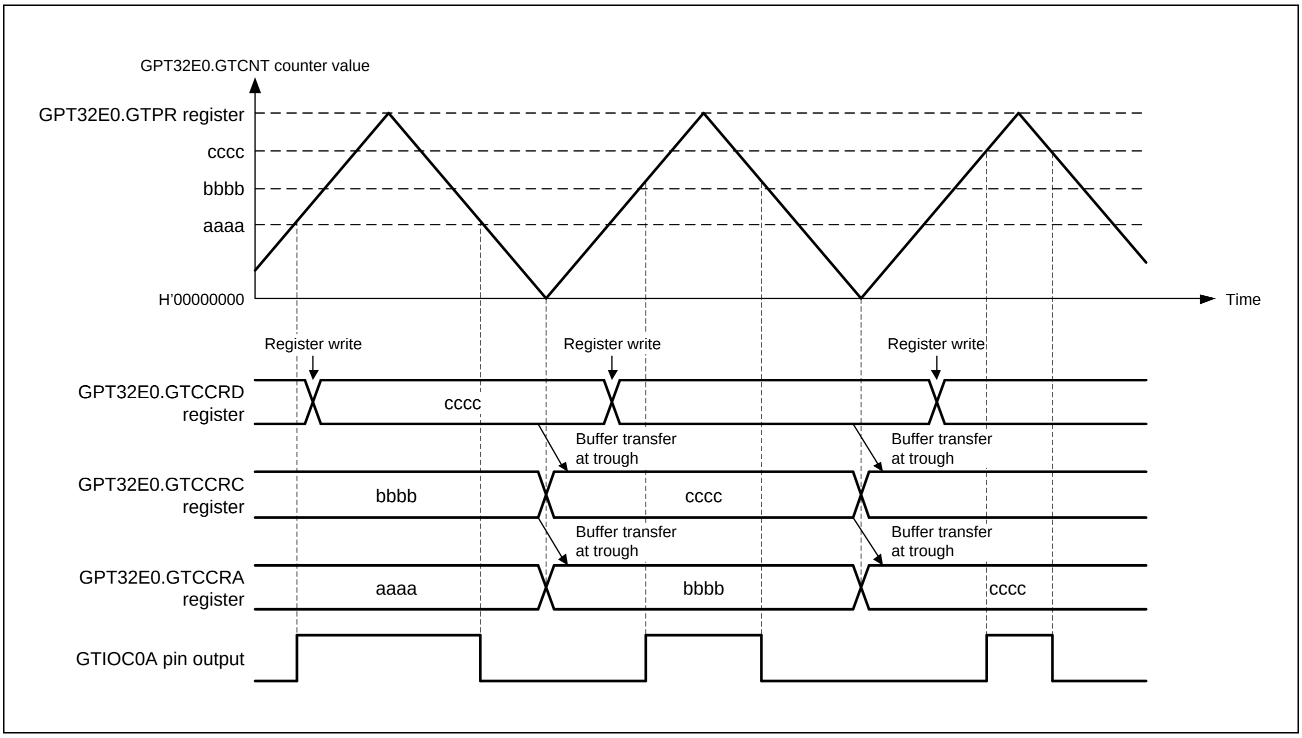


Figure 18.22 Example of GTCCRA and GTCCRB double buffer operation with output compare, triangle waves, buffer operation at trough, output toggled at GTCCRA compare match, and output retained at cycle end

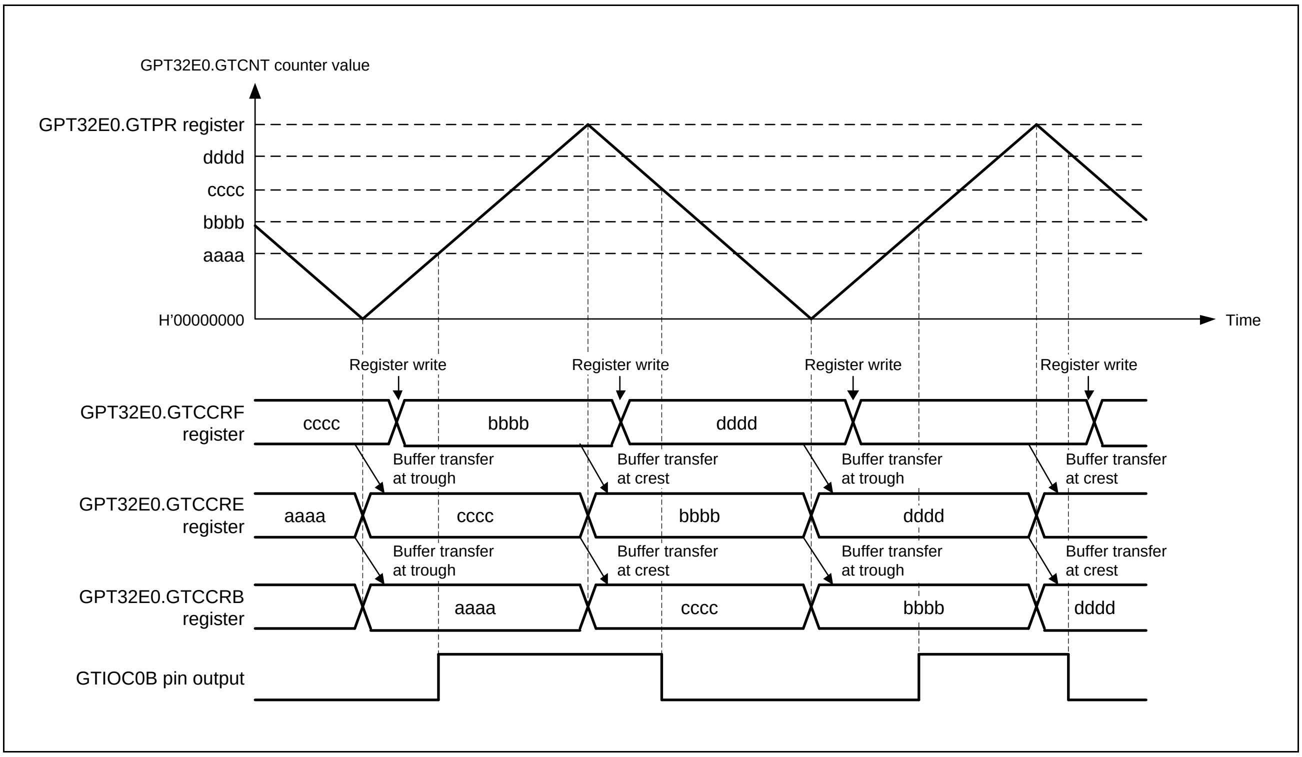
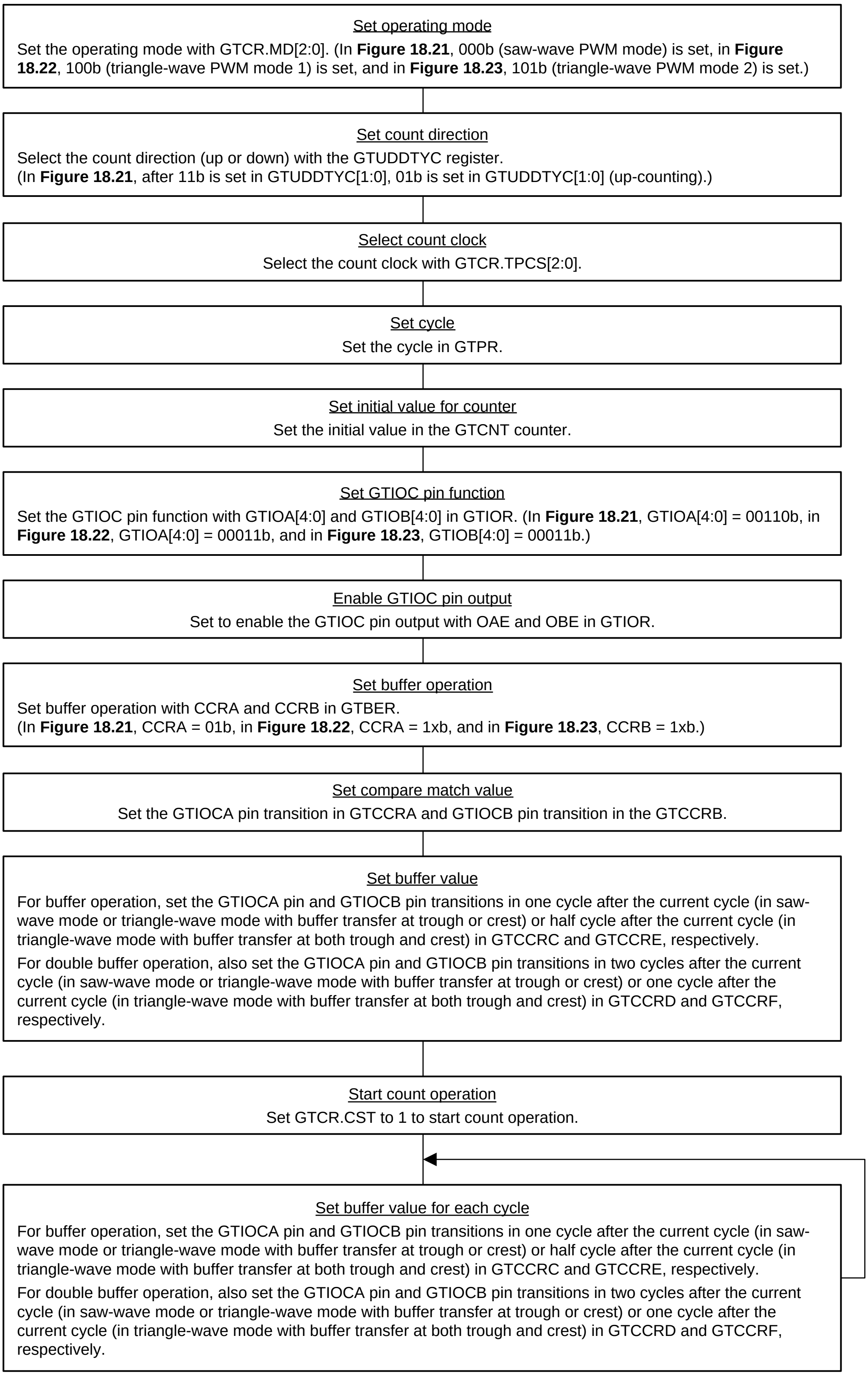


Figure 18.23 Example of GTCCRA and GTCCRB double buffer operation with output compare, triangle waves, buffer operation at both troughs and crests, output toggled at GTCCRB compare match, and output retained at cycle end



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.21**, 000b (saw-wave PWM mode) is set, in **Figure 18.22**, 100b (triangle-wave PWM mode 1) is set, and in **Figure 18.23**, 101b (triangle-wave PWM mode 2) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register.
(In **Figure 18.21**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set GTIOC pin function

Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR. (In **Figure 18.21**, GTIOA[4:0] = 00110b, in **Figure 18.22**, GTIOA[4:0] = 00011b, and in **Figure 18.23**, GTIOB[4:0] = 00011b.)

Enable GTIOC pin output

Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer operation

Set buffer operation with CCRA and CCRB in GTBER.
(In **Figure 18.21**, CCRA = 01b, in **Figure 18.22**, CCRA = 1xb, and in **Figure 18.23**, CCRB = 1xb.)

Set compare match value

Set the GTIOCA pin transition in GTCCRA and GTIOCB pin transition in the GTCCRB.

Set buffer value

For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or half cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTCCRC and GTCCRE, respectively.

For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in two cycles after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or one cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTCCRD and GTCCRF, respectively.

Start count operation

Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle

For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or half cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTCCRC and GTCCRE, respectively.

For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in two cycles after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or one cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTCCRD and GTCCRF, respectively.

Figure 18.24 Example setting for GTCCRA and GTCCRB buffer operation with output compare

(2) When GTCCRA or GTCCRB functions as an input capture register

When an input capture is generated, the GTCNT counter value is transferred to GTCCRA and GTCCRB and the stored GTCCRA and GTCCRB register values are transferred to the buffer registers. In input capture operation, the buffer transfer is not performed by the counter clear.

Figure 18.25 and **Figure 18.26** show examples of GTCCRA and GTCCRB buffer operation and **Figure 18.27** shows an example setting for GTCCRA and GTCCRB buffer operation.

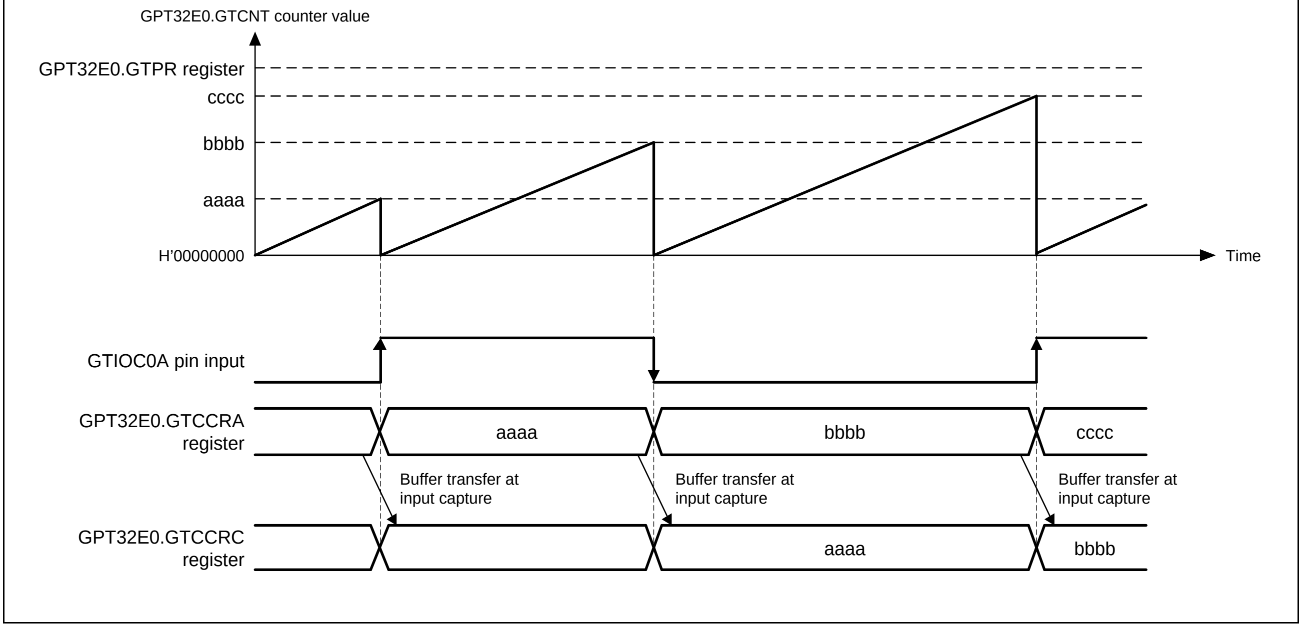


Figure 18.25 Example of GTCCRA and GTCCRB buffer operation with input capture at both edges of GTIOC0A input, saw waves in up-counting, and GTCNT counter cleared at both edges of GTIOC0A input

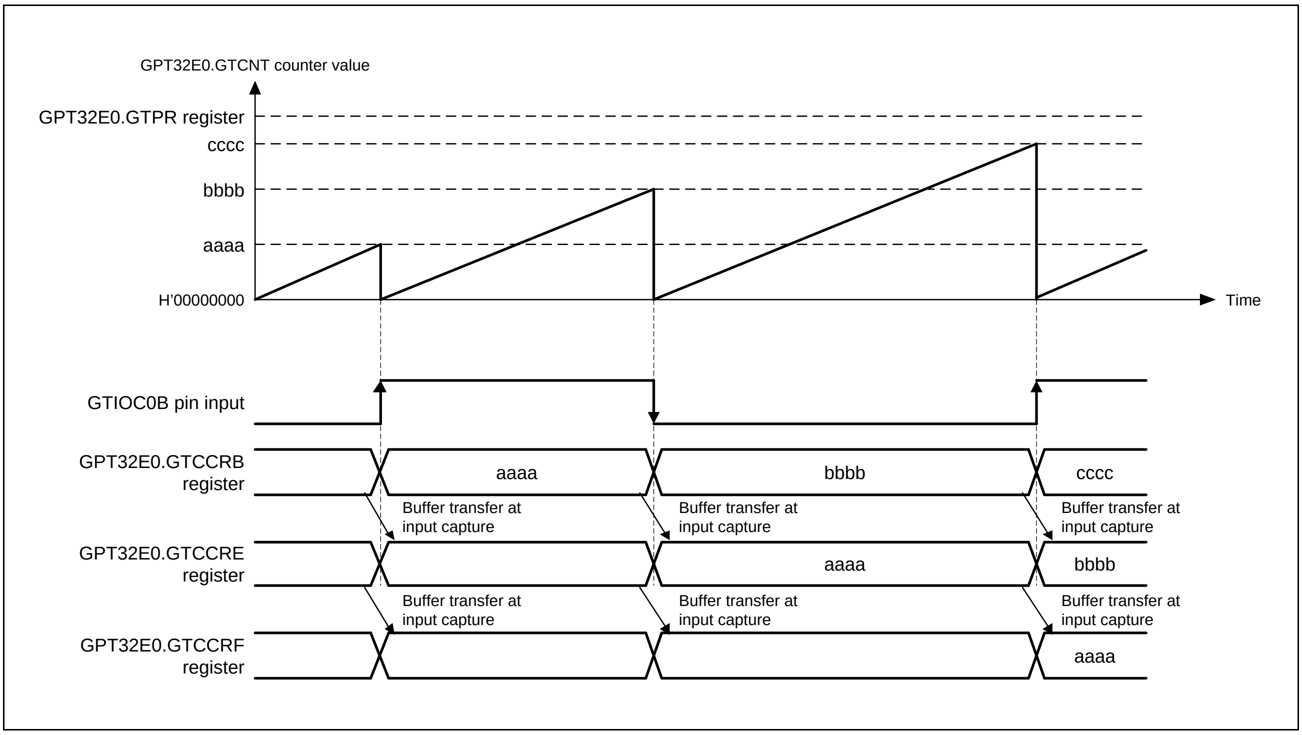
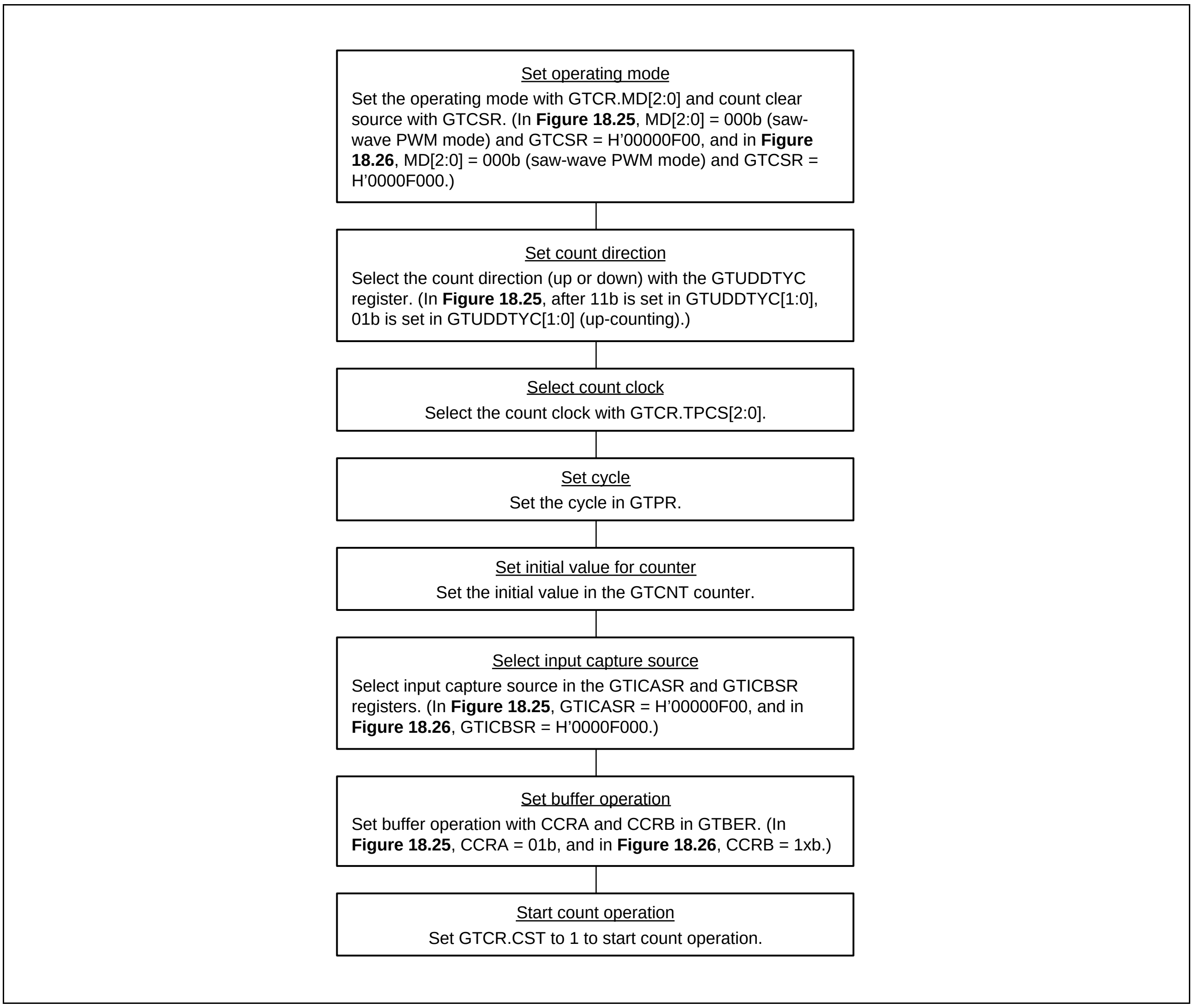


Figure 18.26 Example of GTCCRA and GTCCRB double buffer operation with input capture at both edges of GTIOC0B input, saw waves in up-counting, and GTCNT counter cleared at both edges of GTIOC0B input



Set operating mode

Set the operating mode with GTCR.MD[2:0] and count clear source with GTCSR. (In **Figure 18.25**, MD[2:0] = 000b (saw-wave PWM mode) and GTCSR = H'00000F00, and in **Figure 18.26**, MD[2:0] = 000b (saw-wave PWM mode) and GTCSR = H'0000F000.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.25**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Select input capture source

Select input capture source in the GTICASR and GTICBSR registers. (In **Figure 18.25**, GTICASR = H'00000F00, and in **Figure 18.26**, GTICBSR = H'0000F000.)

Set buffer operation

Set buffer operation with CCRA and CCRB in GTBER. (In **Figure 18.25**, CCRA = 01b, and in **Figure 18.26**, CCRB = 1xb.)

Start count operation

Set GTCR.CST to 1 to start count operation.

Figure 18.27 Example setting for GTCCRA and GTCCRB buffer operation with input capture

18.3.2.3 Buffer Operation for GTADTRA and GTADTRB

GTADTBRA can function as the GTADTRA buffer register and GTADTDBRA can function as the GTADTBRA buffer register (double-buffer register for GTADTRA). Similarly, GTADTBRB can function as the GTADTRB buffer register and GTADTDBRB can function as the GTADTBRB buffer register (double-buffer register for GTADTRB).

To set GTADTRA or GTADTRB to function as a double buffer, set GTBER.ADTDA or GTBER.ADTDB to 1. For single buffer operation, set GTBER.ADTDA or GTBER.ADTDB to 0. To set GTADTRA or GTADTRB to not function as a buffer, set GTBER.ADTTA[1:0] or GTBER.ADTTB[1:0] to 00b.

The buffer transfer timing can be set with the GTBER.ADTTA[1:0] bits. For saw waves, overflows (during up-counting) or underflows (during down-counting) can be selected. For triangle waves, crests are selected when GTBER.ADTTA[1:0] = 01b, troughs are selected when GTBER.ADTTA[1:0] = 10b, and both crests and troughs are selected when GTBER.ADTTA[1:0] = 11b.

Figure 18.28 to **Figure 18.30** show examples of GTADTRA and GTADTRB buffer operation and **Figure 18.31** shows an example setting for GTDTRA and GTADTRB buffer operation.

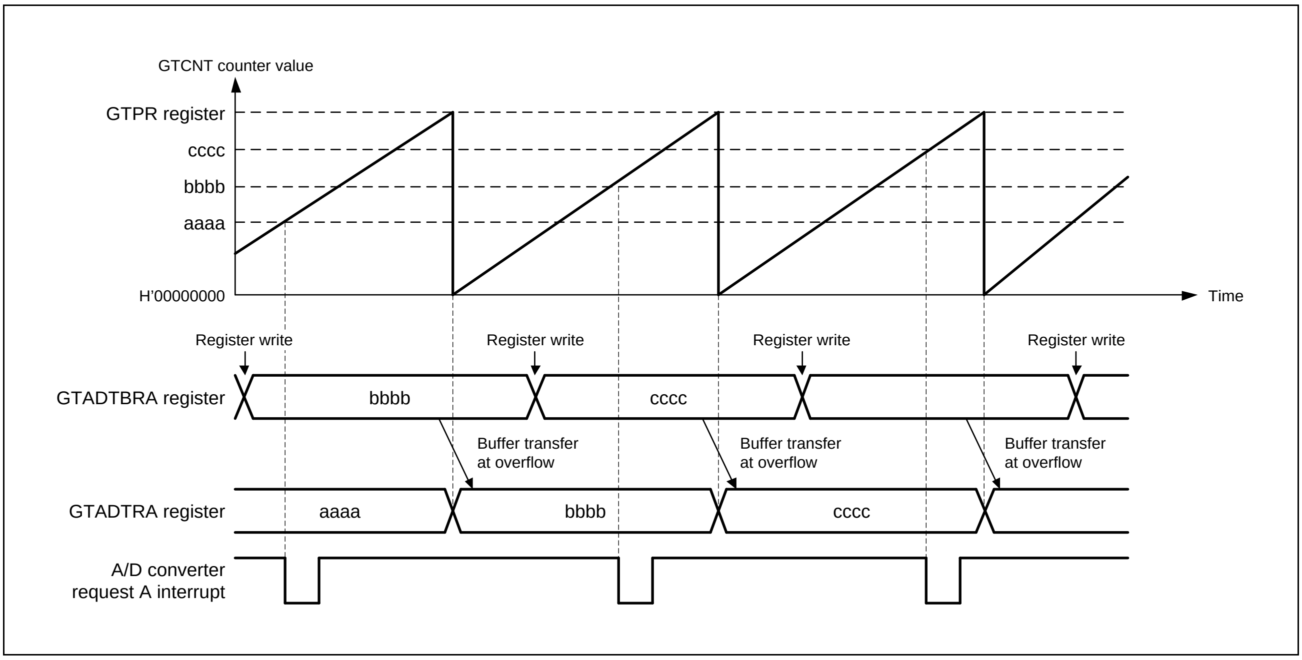


Figure 18.28 Example of GTADTRA and GTADTRB buffer operation with saw waves in up-counting and A/D converter start request interrupt generated by up-counting

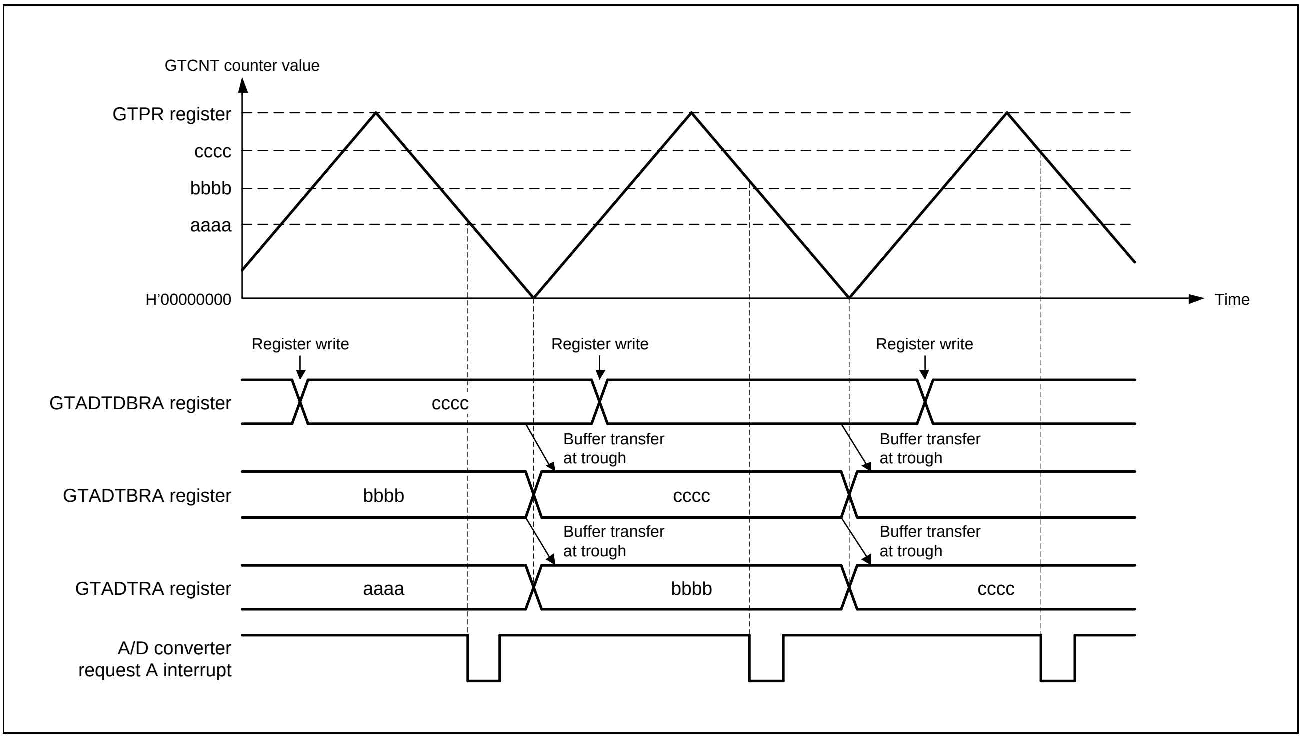


Figure 18.29 Example of GTADTRA and GTADTRB double buffer operation with triangle waves, buffer transfer at troughs, and A/D converter start request interrupt generated by down-counting

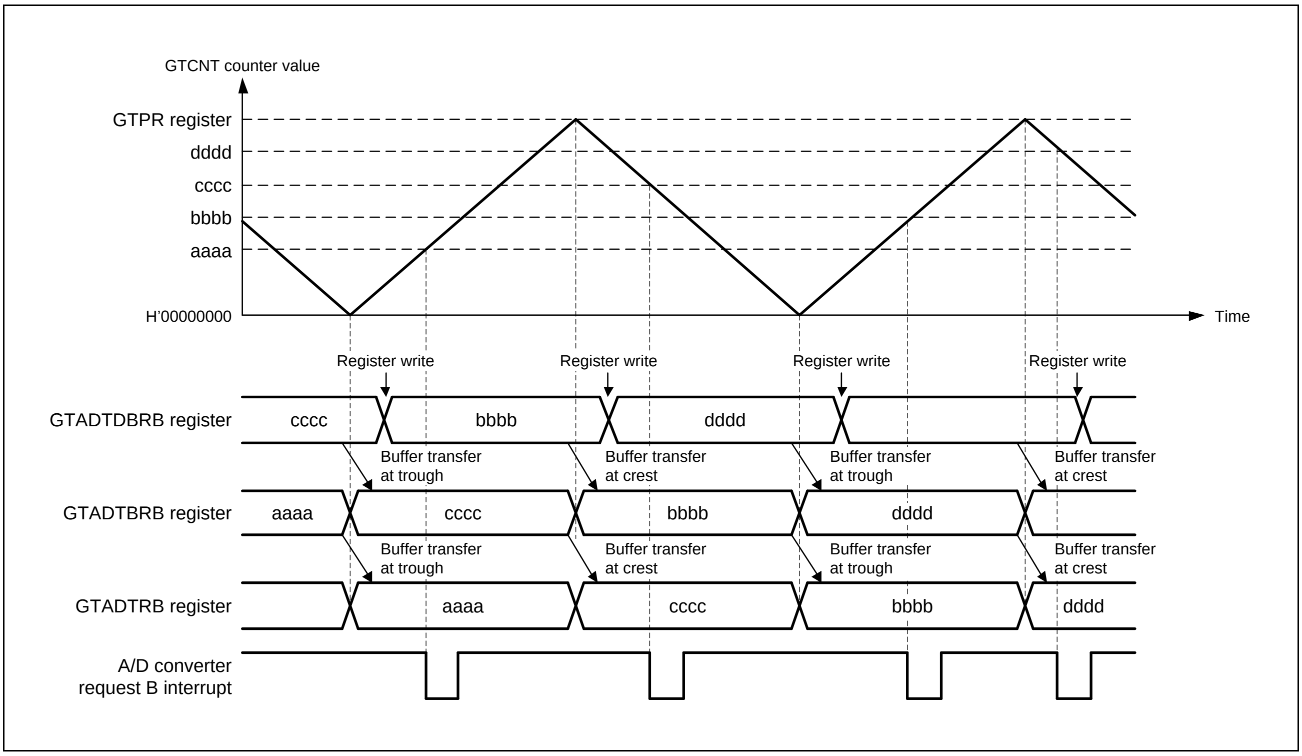
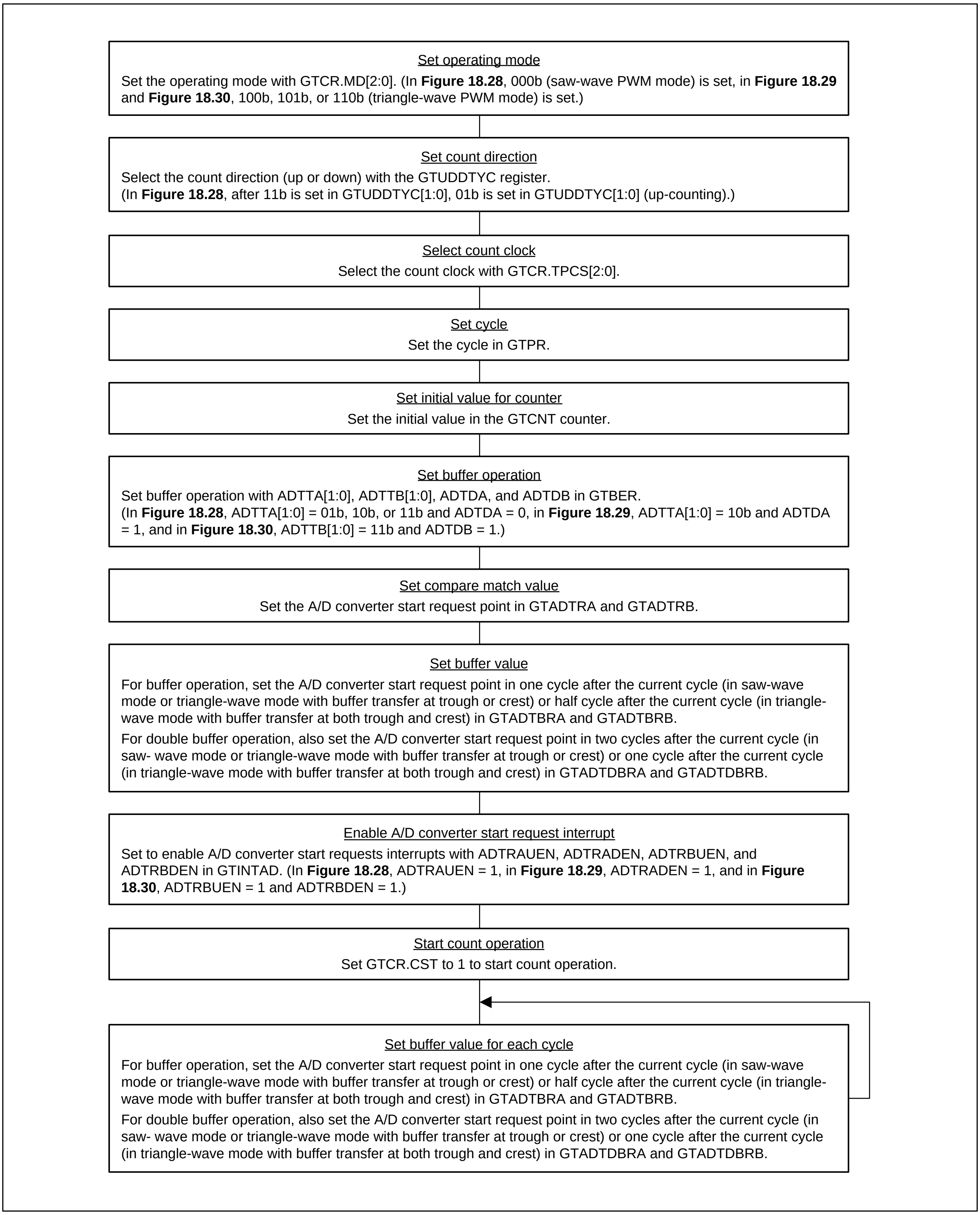


Figure 18.30 Example of GTADTRA and GTADTRB double buffer operation with triangle waves, buffer transfer at both troughs and crests, and A/D converter start request interrupt generated by both up- and down- counting



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.28**, 000b (saw-wave PWM mode) is set, in **Figure 18.29** and **Figure 18.30**, 100b, 101b, or 110b (triangle-wave PWM mode) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register.
(In **Figure 18.28**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set buffer operation

Set buffer operation with ADTTA[1:0], ADTTB[1:0], ADTDA, and ADTDB in GTBER.
(In **Figure 18.28**, ADTTA[1:0] = 01b, 10b, or 11b and ADTDA = 0, in **Figure 18.29**, ADTTA[1:0] = 10b and ADTDA = 1, and in **Figure 18.30**, ADTTB[1:0] = 11b and ADTDB = 1.)

Set compare match value

Set the A/D converter start request point in GTADTRA and GTADTRB.

Set buffer value

For buffer operation, set the A/D converter start request point in one cycle after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or half cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTADTBRA and GTADTBRB.

For double buffer operation, also set the A/D converter start request point in two cycles after the current cycle (in saw- wave mode or triangle-wave mode with buffer transfer at trough or crest) or one cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTADTDBRA and GTADTDBRB.

Enable A/D converter start request interrupt

Set to enable A/D converter start requests interrupts with ADTRAUEN, ADTRADEN, ADTRBUEN, and ADTRBDEN in GTINTAD. (In **Figure 18.28**, ADTRAUEN = 1, in **Figure 18.29**, ADTRADEN = 1, and in **Figure 18.30**, ADTRBUEN = 1 and ADTRBDEN = 1.)

Start count operation

Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle

For buffer operation, set the A/D converter start request point in one cycle after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or half cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTADTBRA and GTADTBRB.

For double buffer operation, also set the A/D converter start request point in two cycles after the current cycle (in saw- wave mode or triangle-wave mode with buffer transfer at trough or crest) or one cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in GTADTDBRA and GTADTDBRB.

Figure 18.31 Example setting for GTADTRA and GTADTRB buffer operation

18.3.3 PWM Output Operating Mode

The GPT can output PWM waveforms to the GTIOCA or GTIOCB pin by a compare match between the GTCNT counter and GTCCRA or GTCCRB.

By setting GTDTCR, GTDVU, and GTDVD, the compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

18.3.3.1 Saw-Wave PWM Mode

In saw-wave PWM mode, GTCNT performs saw-wave (half-wave) operation by setting the cycle in GTPR.A PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

Figure 18.32 shows an example of saw-wave PWM mode operation, and **Figure 18.33** shows an example setting for saw-wave PWM mode.

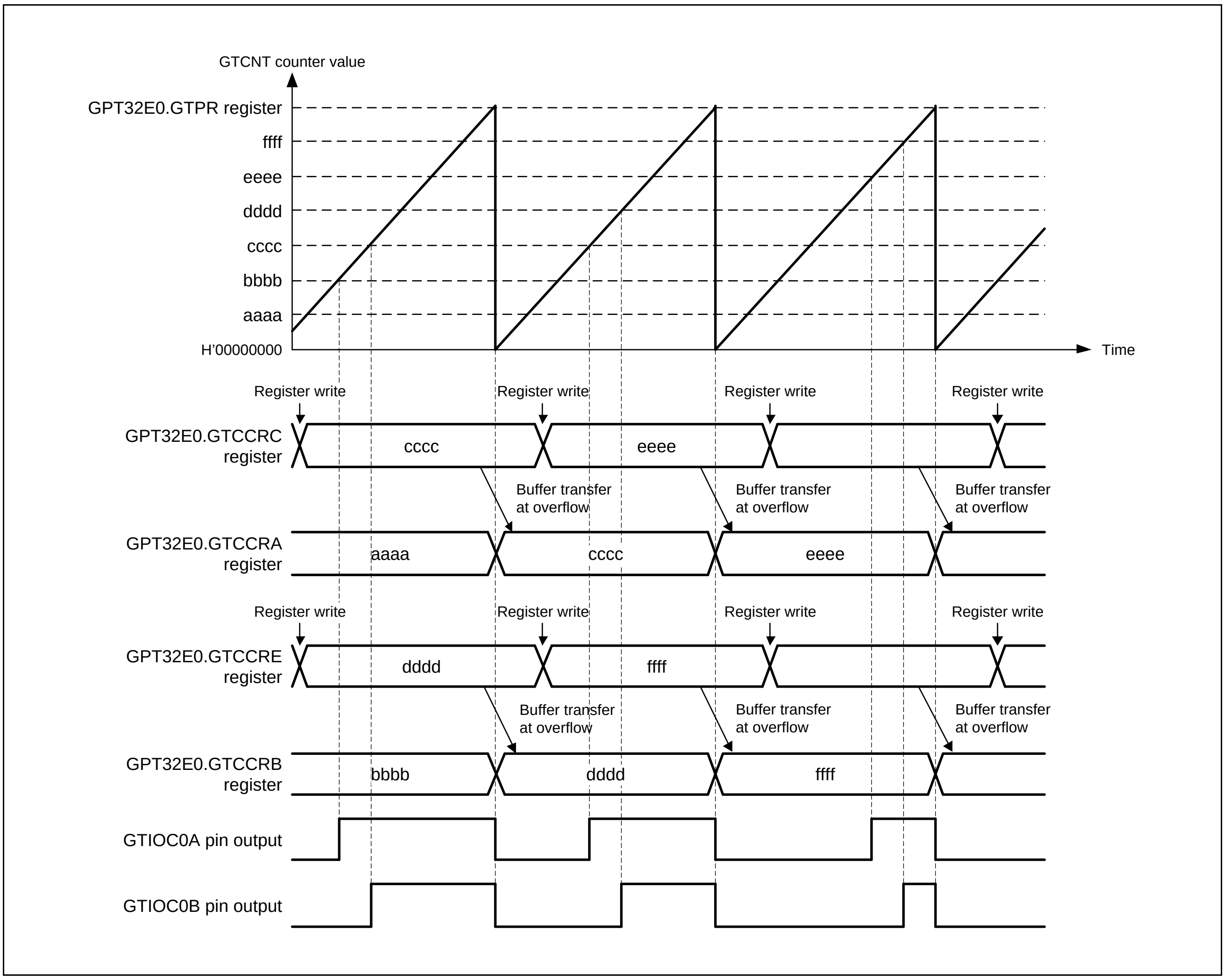
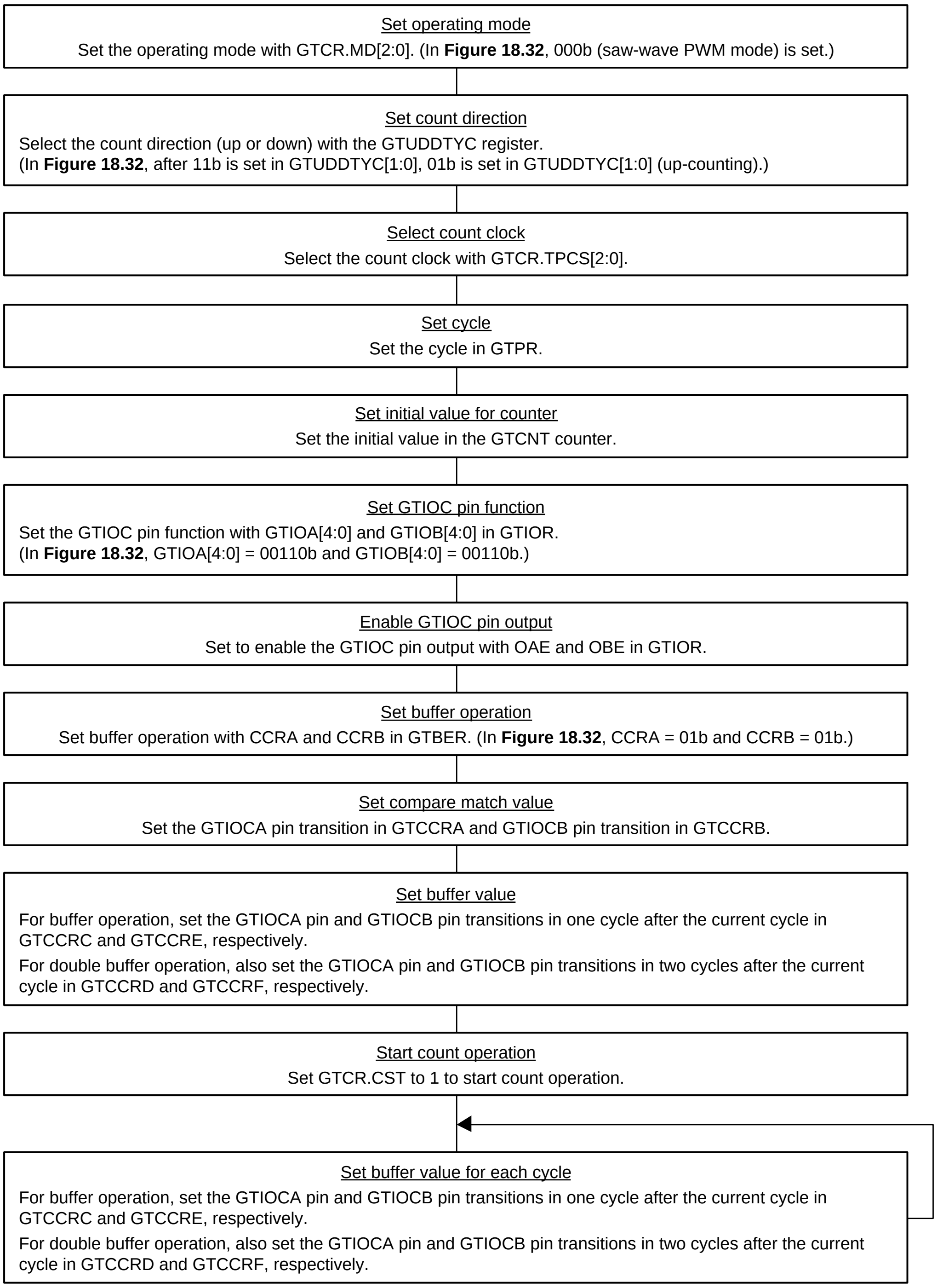


Figure 18.32 Example of saw-wave PWM mode operation with up-counting, buffer operation, high output at GTCCRA/GTCCRB compare match, and low output at cycle end



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.32**, 000b (saw-wave PWM mode) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register.
(In **Figure 18.32**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set GTIOC pin function

Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR.
(In **Figure 18.32**, GTIOA[4:0] = 00110b and GTIOB[4:0] = 00110b.)

Enable GTIOC pin output

Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer operation

Set buffer operation with CCRA and CCRB in GTBER. (In **Figure 18.32**, CCRA = 01b and CCRB = 01b.)

Set compare match value

Set the GTIOCA pin transition in GTCCRA and GTIOCB pin transition in GTCCRB.

Set buffer value

For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle in GTCCRC and GTCCRE, respectively.

For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in two cycles after the current cycle in GTCCRD and GTCCRF, respectively.

Start count operation

Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle

For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle in GTCCRC and GTCCRE, respectively.

For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in two cycles after the current cycle in GTCCRD and GTCCRF, respectively.

Figure 18.33 Example setting for saw-wave PWM mode

18.3.3.2 Saw-Wave One-Shot Pulse Mode

The saw-wave one-shot pulse mode is a mode in which the cycle is set in GTPR. The GTCNT counter performs saw-wave (half-wave) operation and a PWM waveform is output to the GTIOCA or GTIOCB pin at a compare match of GTCCRA or GTCCRB with buffer operation fixed.

Buffer operation in saw-wave one-shot pulse mode is different from the usual buffer operation. Buffer transfer is performed from the following:

- GTCCRC to GTCCRA at the cycle end
- GTCCRE to GTCCRB at the cycle end
- GTCCRD to temporary register A at the cycle end
- GTCCRF to temporary register B at the cycle end
- Temporary register A to GTCCRA at a GTCCRA compare match
- Temporary register B to GTCCRB at a GTCCRB compare match.

The pin output value can be selected from low output, high output, or toggle output separately for a compare match and the cycle end based on the GTIOR setting.

When 1 is written to the GTBER.CCRSWT bit while counting is stopped, the values of the GTCCRD and GTCCRF registers are forcibly transferred to the temporary registers A and B, respectively. By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.34 shows an example of saw-wave one-shot pulse mode operation, and **Figure 18.35** shows an example setting for saw-wave one-shot pulse mode.

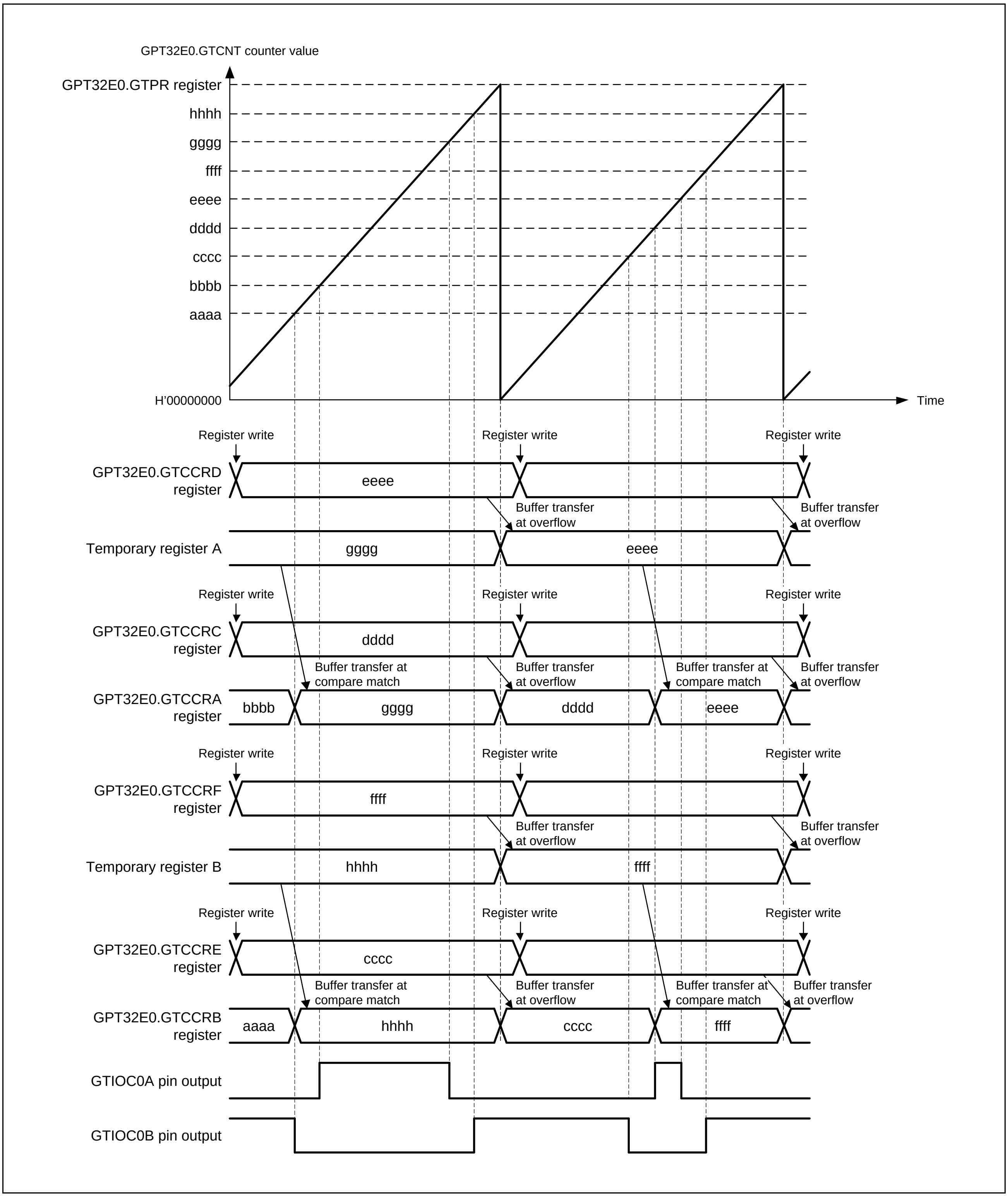
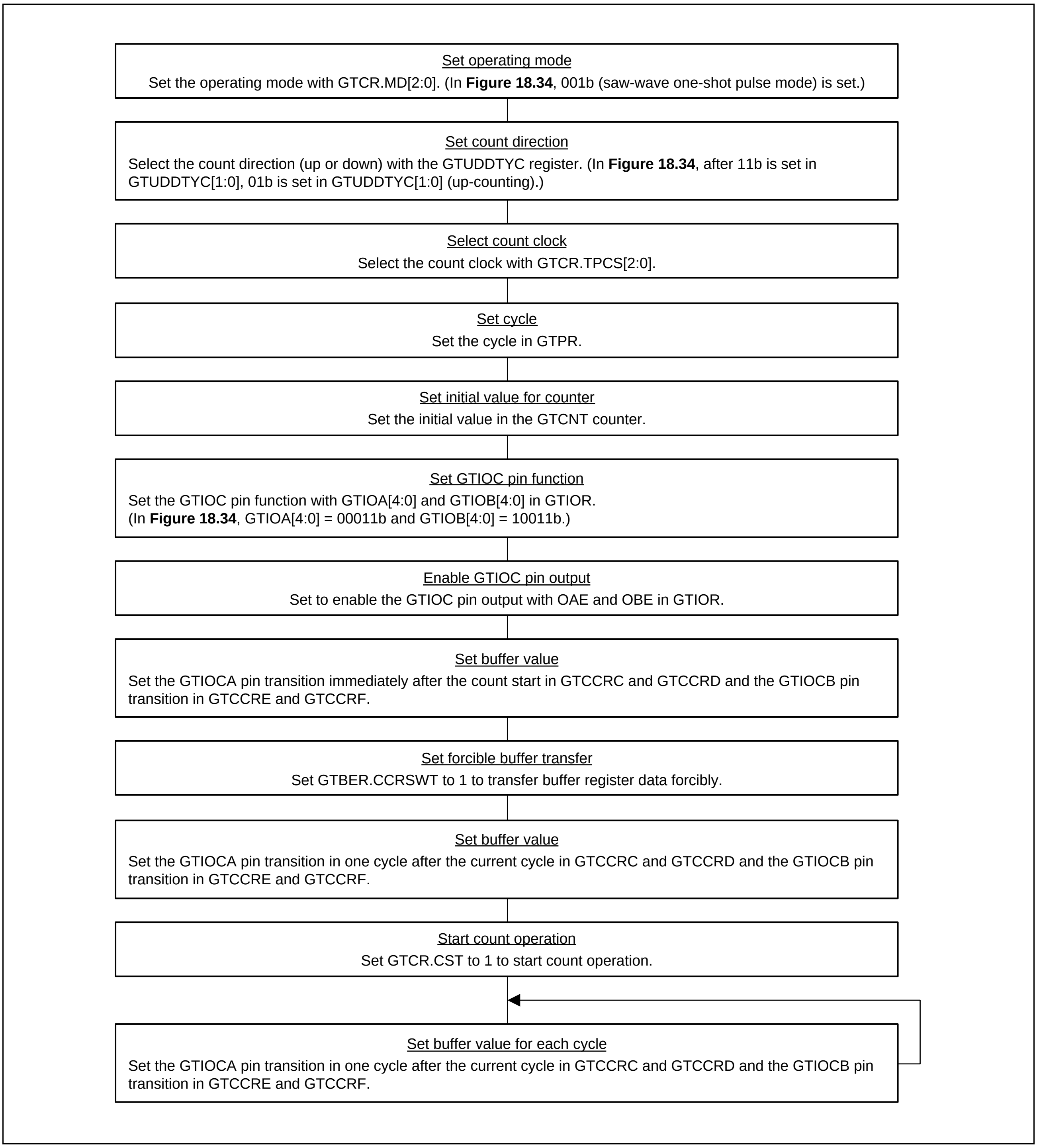


Figure 18.34 Example of saw-wave one-shot pulse mode operation with up-counting, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end



Set operating mode
Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.34**, 001b (saw-wave one-shot pulse mode) is set.)

Set count direction
Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.34**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.

Set GTIOC pin function
Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR.
(In **Figure 18.34**, GTIOA[4:0] = 00011b and GTIOB[4:0] = 10011b.)

Enable GTIOC pin output
Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer value
Set the GTIOCA pin transition immediately after the count start in GTCCRC and GTCCRD and the GTIOCB pin transition in GTCCRE and GTCCRF.

Set forcible buffer transfer
Set GTBER.CCRSWT to 1 to transfer buffer register data forcibly.

Set buffer value
Set the GTIOCA pin transition in one cycle after the current cycle in GTCCRC and GTCCRD and the GTIOCB pin transition in GTCCRE and GTCCRF.

Start count operation
Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle
Set the GTIOCA pin transition in one cycle after the current cycle in GTCCRC and GTCCRD and the GTIOCB pin transition in GTCCRE and GTCCRF.

Figure 18.35 Example setting for saw-wave one-shot pulse mode

18.3.3.3 Triangle-Wave PWM Mode 1 (32-bit transfer at trough)

The triangle-wave PWM mode 1 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. Buffer transfer is performed at the trough. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.36 shows an example of a triangle-wave PWM mode 1 operation, and **Figure 18.37** shows an example setting for a triangle-wave PWM mode 1.

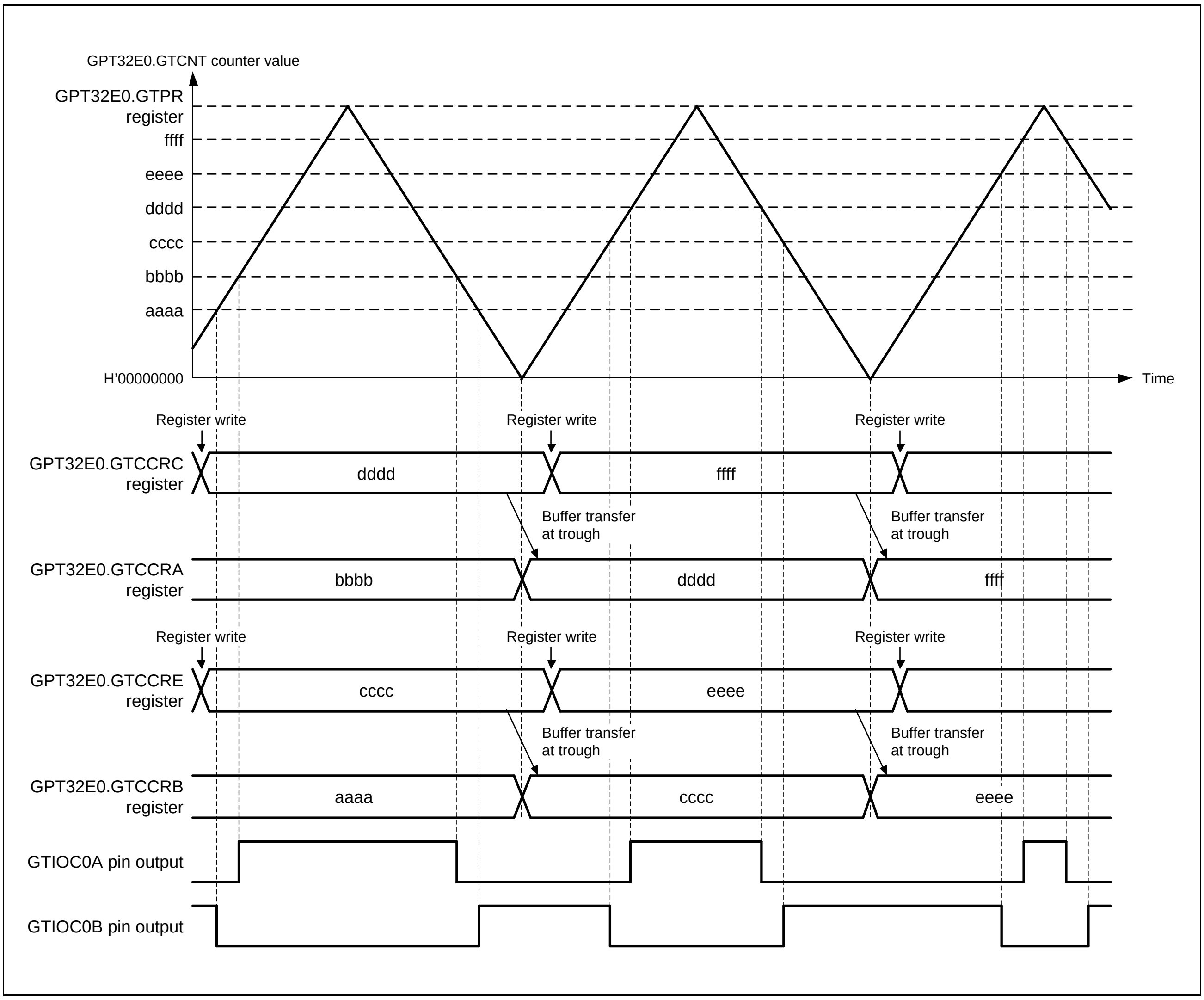
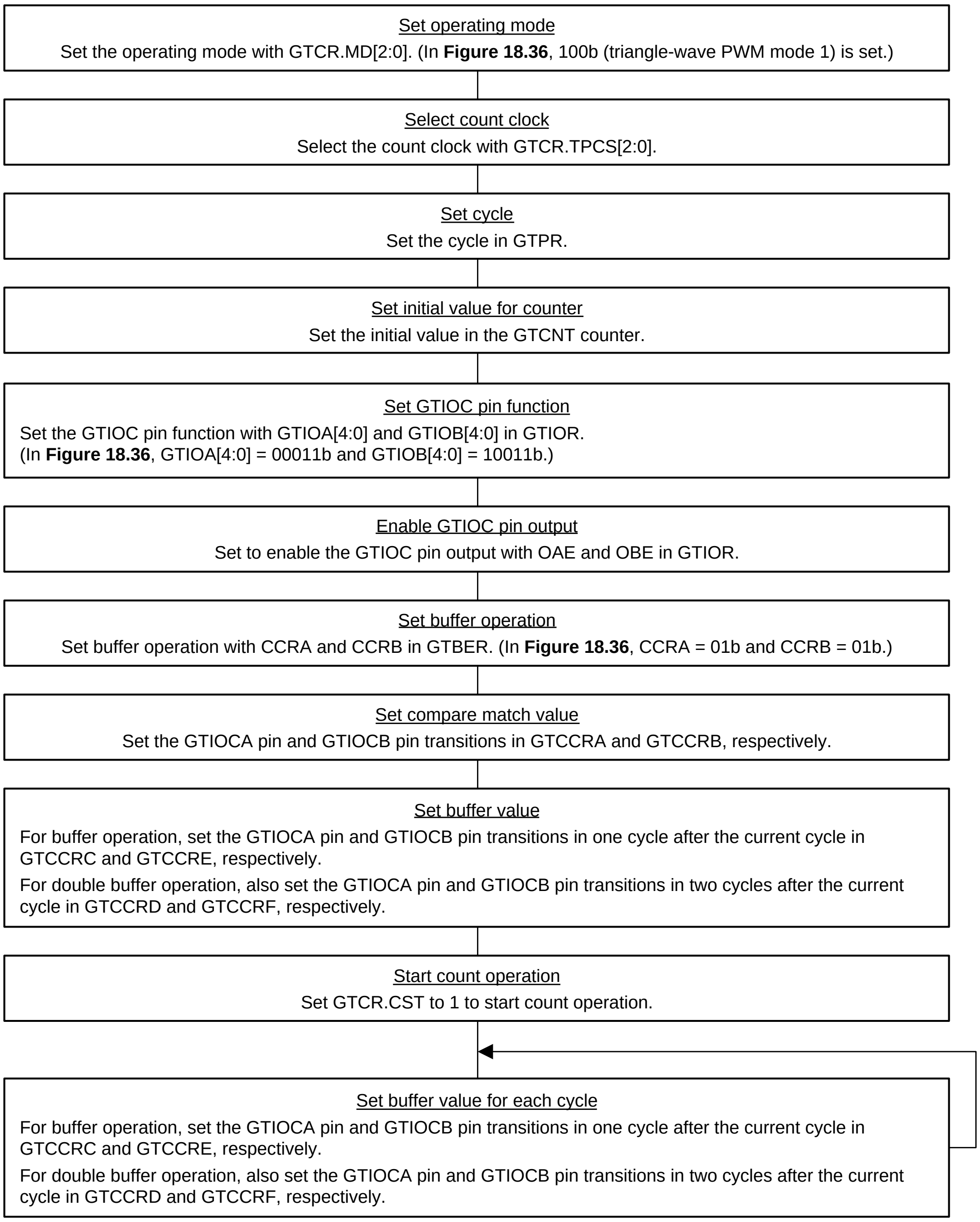


Figure 18.36 Example of triangle-wave PWM mode 1 operation with buffer operation, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB register compare match, and output retained at cycle end



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.36**, 100b (triangle-wave PWM mode 1) is set.)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set GTIOC pin function

Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR. (In **Figure 18.36**, GTIOA[4:0] = 00011b and GTIOB[4:0] = 10011b.)

Enable GTIOC pin output

Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer operation

Set buffer operation with CCRA and CCRB in GTBER. (In **Figure 18.36**, CCRA = 01b and CCRB = 01b.)

Set compare match value

Set the GTIOCA pin and GTIOCB pin transitions in GTCCRA and GTCCRB, respectively.

Set buffer value

For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle in GTCCRC and GTCCRE, respectively.

For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in two cycles after the current cycle in GTCCRD and GTCCRF, respectively.

Start count operation

Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle

For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle in GTCCRC and GTCCRE, respectively.

For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in two cycles after the current cycle in GTCCRD and GTCCRF, respectively.

Figure 18.37 Example setting for triangle-wave PWM mode 1

18.3.3.4 Triangle-Wave PWM Mode 2 (32-bit transfer at crest and trough)

Similarly to triangle-wave PWM mode 1, in triangle-wave PWM mode 2 the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCA or GTIOCB pin when a GTCCRA or GTCCRB compare match occurs. The buffer transfer is performed at both crests and troughs. The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.38 shows an example of triangle-wave PWM mode 2 operation, and **Figure 18.39** shows an example setting for triangle-wave PWM mode 2.

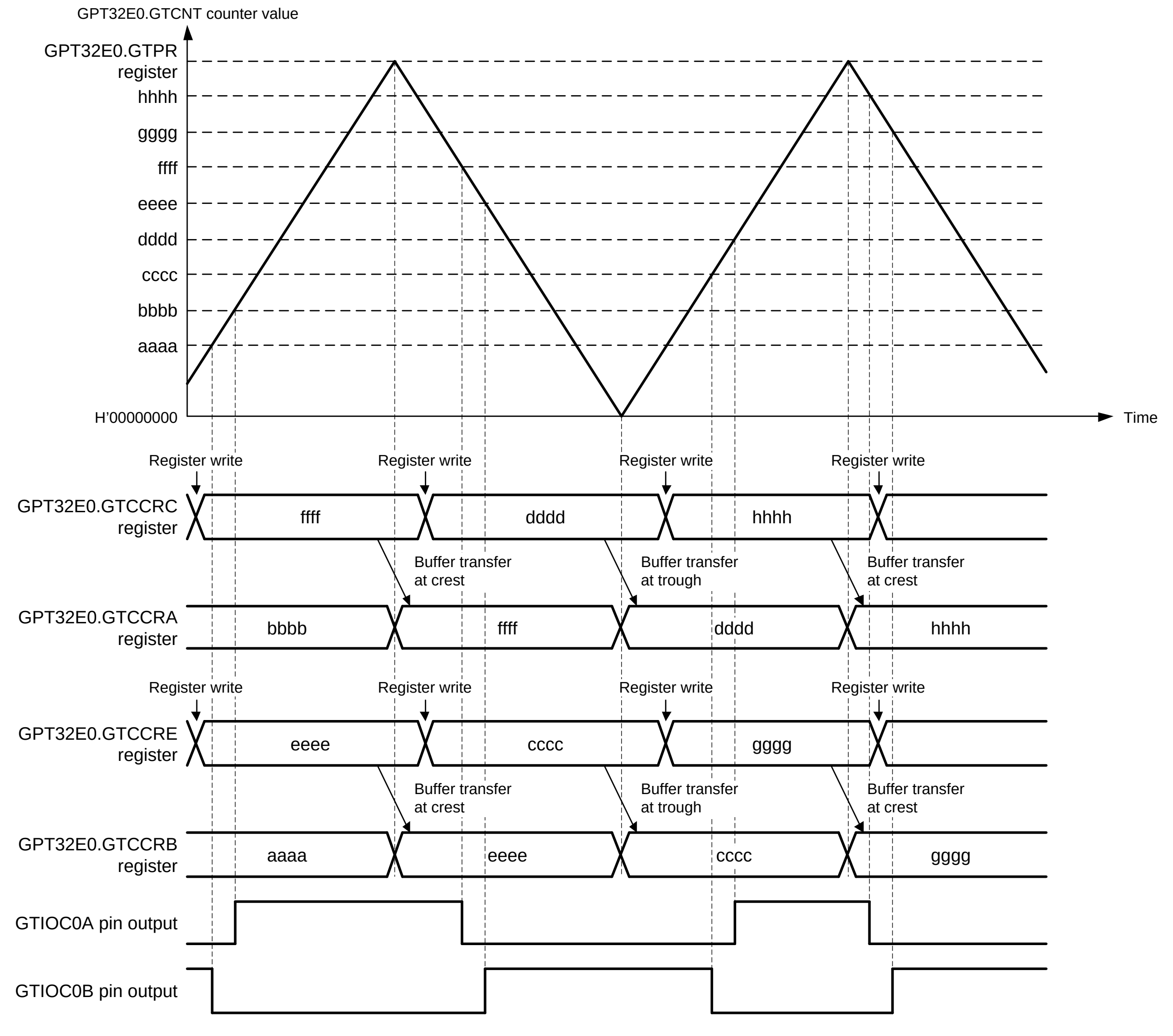
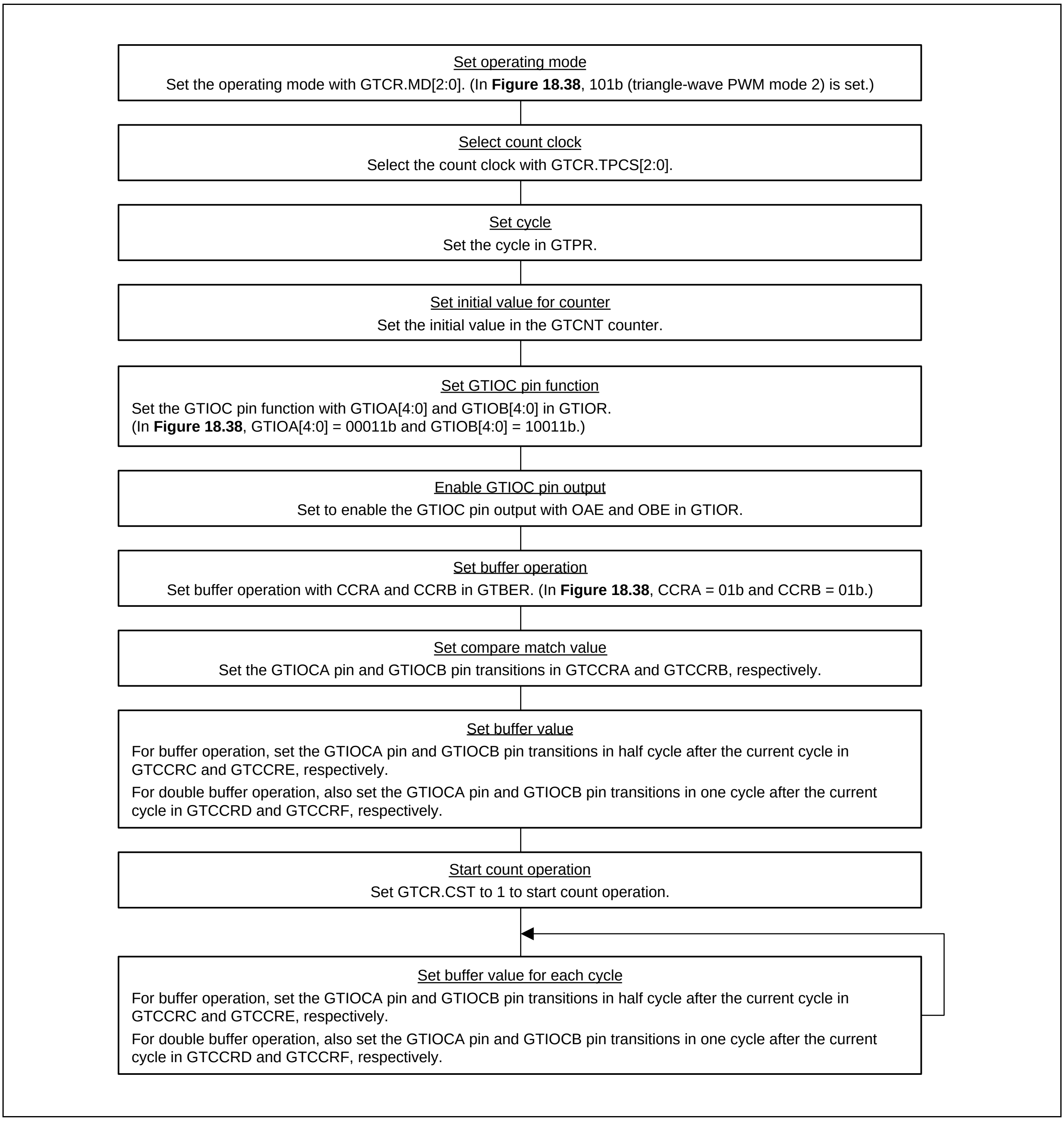


Figure 18.38 Example of triangle-wave PWM mode 2 operation with buffer operation, low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end



Set operating mode
Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.38**, 101b (triangle-wave PWM mode 2) is set.)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.

Set GTIOC pin function
Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR.
(In **Figure 18.38**, GTIOA[4:0] = 00011b and GTIOB[4:0] = 10011b.)

Enable GTIOC pin output
Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer operation
Set buffer operation with CCRA and CCRB in GTBER. (In **Figure 18.38**, CCRA = 01b and CCRB = 01b.)

Set compare match value
Set the GTIOCA pin and GTIOCB pin transitions in GTCCRA and GTCCRB, respectively.

Set buffer value
For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in half cycle after the current cycle in GTCCRC and GTCCRE, respectively.
For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle in GTCCRD and GTCCRF, respectively.

Start count operation
Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle
For buffer operation, set the GTIOCA pin and GTIOCB pin transitions in half cycle after the current cycle in GTCCRC and GTCCRE, respectively.
For double buffer operation, also set the GTIOCA pin and GTIOCB pin transitions in one cycle after the current cycle in GTCCRD and GTCCRF, respectively.

Figure 18.39 Example setting for triangle-wave PWM mode 2

18.3.3.5 Triangle-Wave PWM Mode 3 (64-bit transfer at trough)

The triangle-wave PWM mode 3 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation and a PWM waveform is output to the GTIOCA or GTIOCB pin at a compare match of GTCCRA or GTCCRB with buffer operation fixed. Buffer operation in triangle-wave PWM mode 3 is different from the usual buffer operation. Buffer transfer is performed from the following:

- GTCCRC to GTCCRA at the trough
- GTCCRE to GTCCRB at the trough
- GTCCRD to temporary register A at the trough
- GTCCRF to temporary register B at the trough
- Temporary register A to GTCCRA at the crest
- Temporary register B to GTCCRB at the crest

The pin output value can be selected from low output, high output, or toggle output separately for a compare match and for the cycle end based on the GTIOR setting. By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can automatically be set to GTCCRB.

Figure 18.40 shows an example of triangle-wave PWM mode 3 operation, and **Figure 18.41** shows an example setting for triangle-wave PWM mode 3.

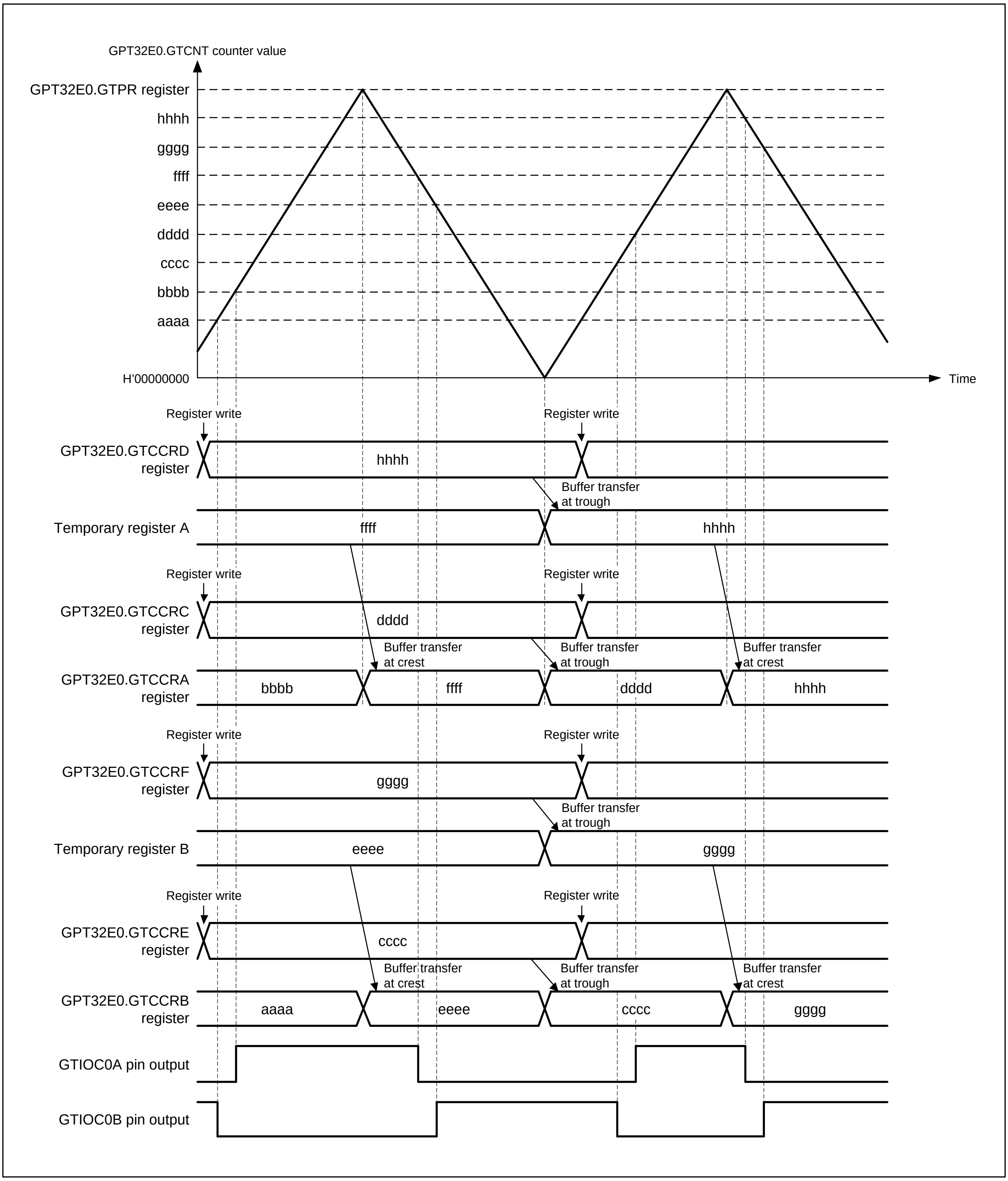
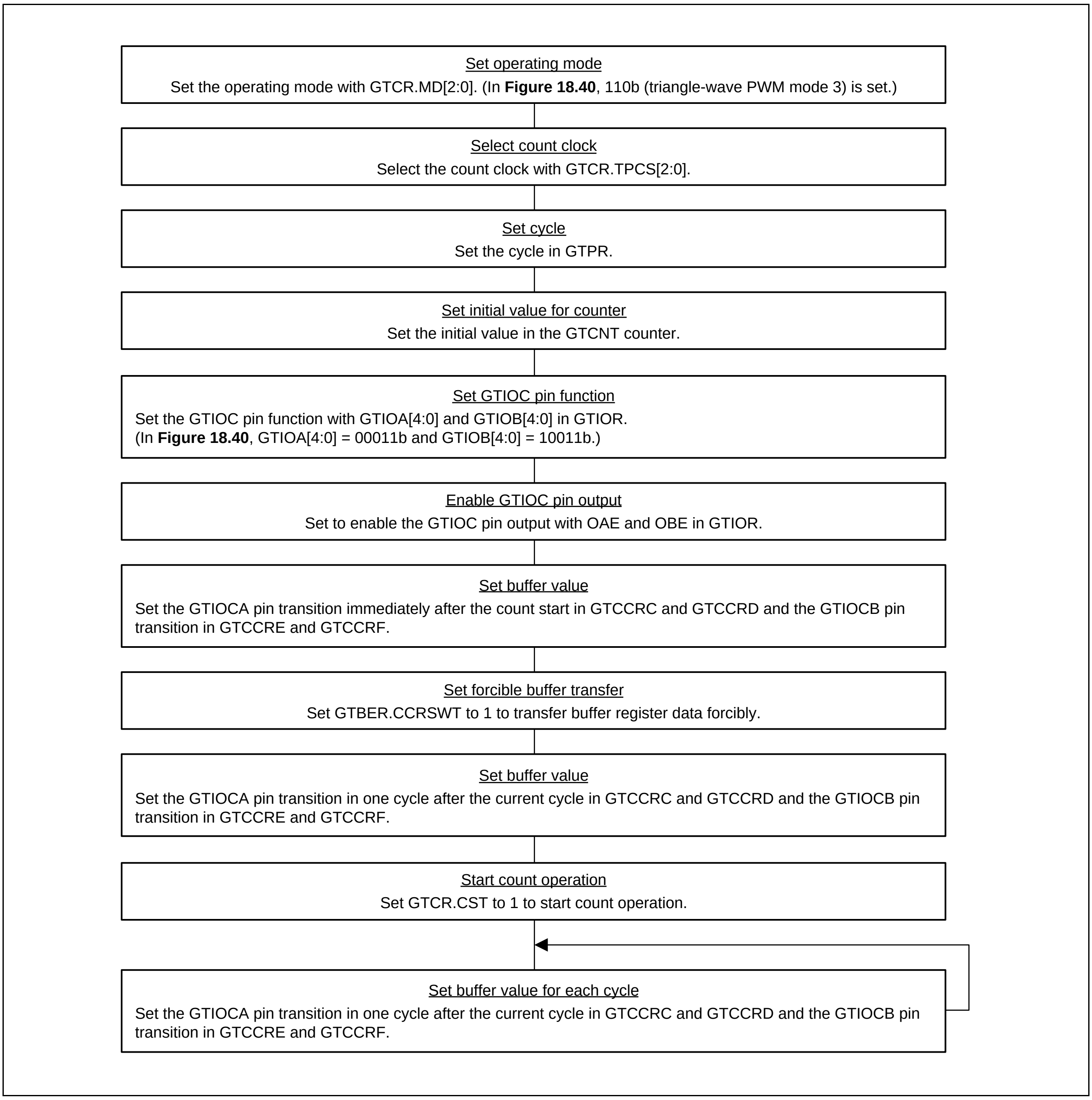


Figure 18.40 Example of triangle-wave PWM mode 3 operation with low output from the GTIOC0A pin and high output from the GTIOC0B pin at count start, output toggled at GTCCRA/GTCCRB compare match, and output retained at cycle end



Set operating mode
Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.40**, 110b (triangle-wave PWM mode 3) is set.)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.

Set GTIOC pin function
Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR.
(In **Figure 18.40**, GTIOA[4:0] = 00011b and GTIOB[4:0] = 10011b.)

Enable GTIOC pin output
Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer value
Set the GTIOCA pin transition immediately after the count start in GTCCRC and GTCCRD and the GTIOCB pin transition in GTCCRE and GTCCRF.

Set forcible buffer transfer
Set GTBER.CCRSWT to 1 to transfer buffer register data forcibly.

Set buffer value
Set the GTIOCA pin transition in one cycle after the current cycle in GTCCRC and GTCCRD and the GTIOCB pin transition in GTCCRE and GTCCRF.

Start count operation
Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle
Set the GTIOCA pin transition in one cycle after the current cycle in GTCCRC and GTCCRD and the GTIOCB pin transition in GTCCRE and GTCCRF.

Figure 18.41 Example setting for triangle-wave PWM mode 3

18.3.4 Automatic Dead Time Setting Function

By setting GTDTCR, a compare match value for a negative waveform with dead time obtained by a compare match value for a positive waveform (GTCCRA value) and specified dead time values (GTDVU and GTDVD values) can automatically be set to GTCCRB. The automatic dead time setting function can be used in saw-wave one-shot pulse mode and all the triangle PWM modes.

Dead time can be separately set for the first half and second half of a waveform. Dead time for the transition in the first half of a negative waveform is set in GTDVU and that in the second half is set in GTDVD. The same dead time can also be set for the first and second halves by setting the GTDTCR.TDFER bit to 1.

GTDBU can be used as a buffer register of GTDVU, and GTDBD can be used as a buffer register of GTDVD.

In the saw-wave mode, buffer transfer is performed when the GTCND counter overflows (up-counting), underflows (down-counting), or is cleared. In the triangle-wave mode, transfer proceeds at troughs.

The change point of the negative phase waveform with the automatic dead time setting can be confirmed by reading the GTCCRB register value.

Writing to the GTCCRB register is prohibited when using the automatic dead time setting function. It is prohibited to set dead time such that the change point of the waveform exceeds the count cycle.

If there is an error condition on the dead time setting, be able to generate waveforms that secured dead time by correcting the transition points of the positive and negative phase waveforms, as shown in **Table 18.6**.

The transition point of the corrected negative-phase waveform is automatically set in the GTCCRB register. Since the internal signal is used to judge the transition point of the positive phase waveform, the GTCCRA register is not updated with the corrected value.

By correcting the waveform transition point due to a dead time error on the saw-wave one-shot pulse mode, if the order of waveform transition points is disturbed or it exceeds the count cycle period after correction, the complementary relationship between the positive-phase waveform and the negative-phase waveform is not guaranteed.

By setting the GTCCRA equal 0 or greater than or equal to the GTPR of the GTCCRA register in the triangle-wave PWM mode, if the dead time setting exceeds the count cycle period, output transition is suppressed by the output protection function. When GTCCRA ≥ GTPR + GTDVn, GTPR - 1 is set as the upper limit value in the GTCCRB register (Refer to **Section 18.7.4, Output Protection Function for GTIOC Pin Output**).

When GTCCRA ≥ GTPR + GTDVn, GTPR - 1 is set as the upper limit value in the GTCCRB register.

Table 18.6 Correction of waveform transition point at dead time error occurrence

Wave mode	Count direction	Interval	Dead time error condition	Positive-Phase waveform transition point with corrected	Negative-phase waveform transition point with corrected
saw-wave one-shot pulse mode	Up counting	first half	GTCCRA-GTDVU < 0	GTDVU	0
		second half	GTCCRA-GTDVD > GTPR	GTPR-GTDVD	GTPR
	Down counting	first half	GTCCRA-GTDVU > GTPR	GTPR-GTDVU	GTPR
		second half	GTCCRA-GTDVD < 0	GTDVD	0
triangle PWM mode 1/2/3	Up counting	(first half)	GTCCRA-GTDVU ≤ 0	GTDVU+1	1
	Down counting	(second half)	GTCCRA-GTDVD < 0	GTDVD	0

Values for automatic dead time setting can be read from GTCCRB. The automatic dead time value setting to GTCCRB is performed at the next count clock cycle when registers that are used for calculating the automatic dead time value are updated.

The method of writing a new value to a GTDVn register depends on whether buffer operation is enabled or disabled. When GTDVn buffer operation is enabled: The GTDBn register can be written at any time. The value in the GTDBn register is transferred to the GTDVn register at the cycle end.

When GTDVn buffer operation is disabled: Set the GTCR.CST bit to stop the GPT before changing the value of the GTDVn register.

Figure 18.42 to **Figure 18.45** show examples of automatic dead time setting function operation. **Figure 18.46** and **Figure 18.47** show the setting examples.

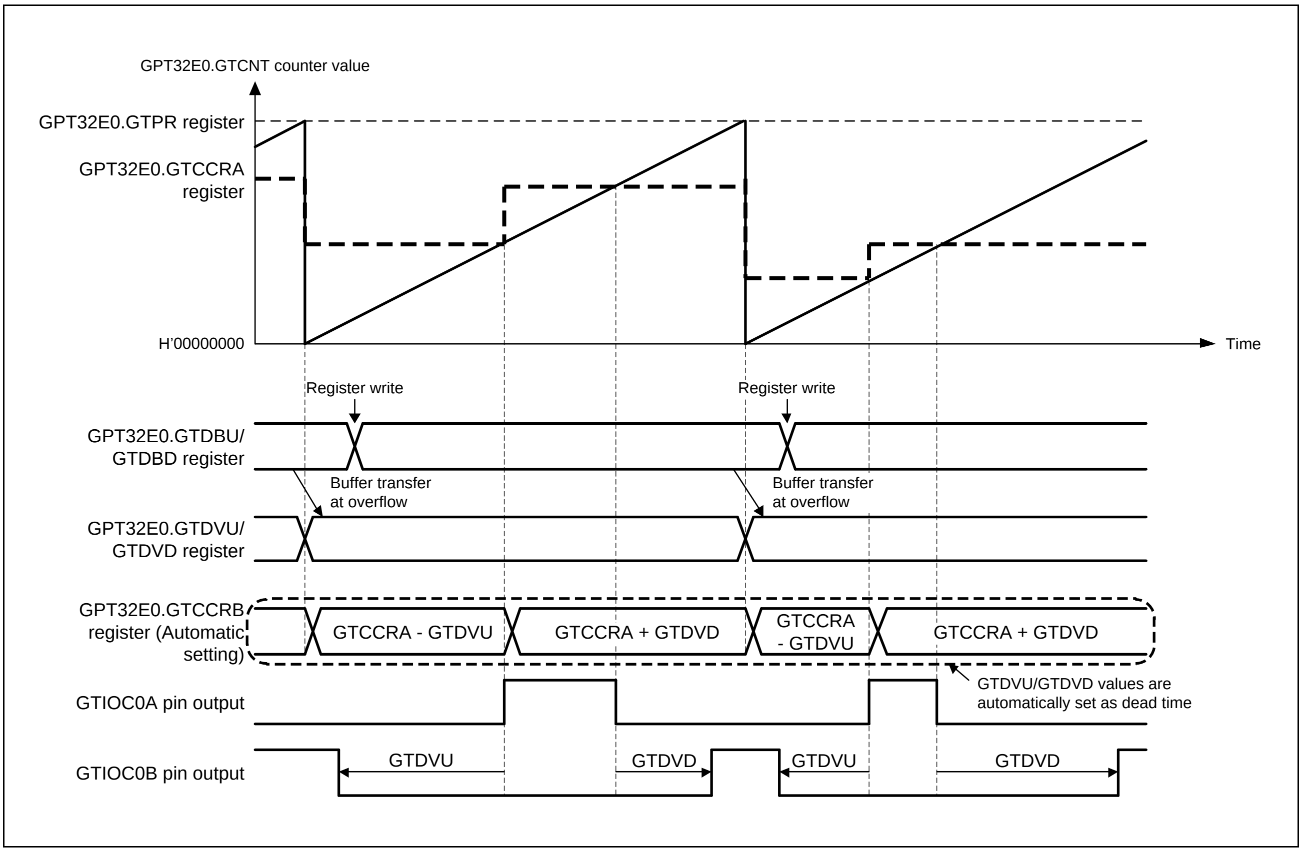


Figure 18.42 Example of automatic dead time setting function operation with saw-wave one-shot pulse mode, up-counting, GTDVU and GTDVD set to buffer operation, and active-high

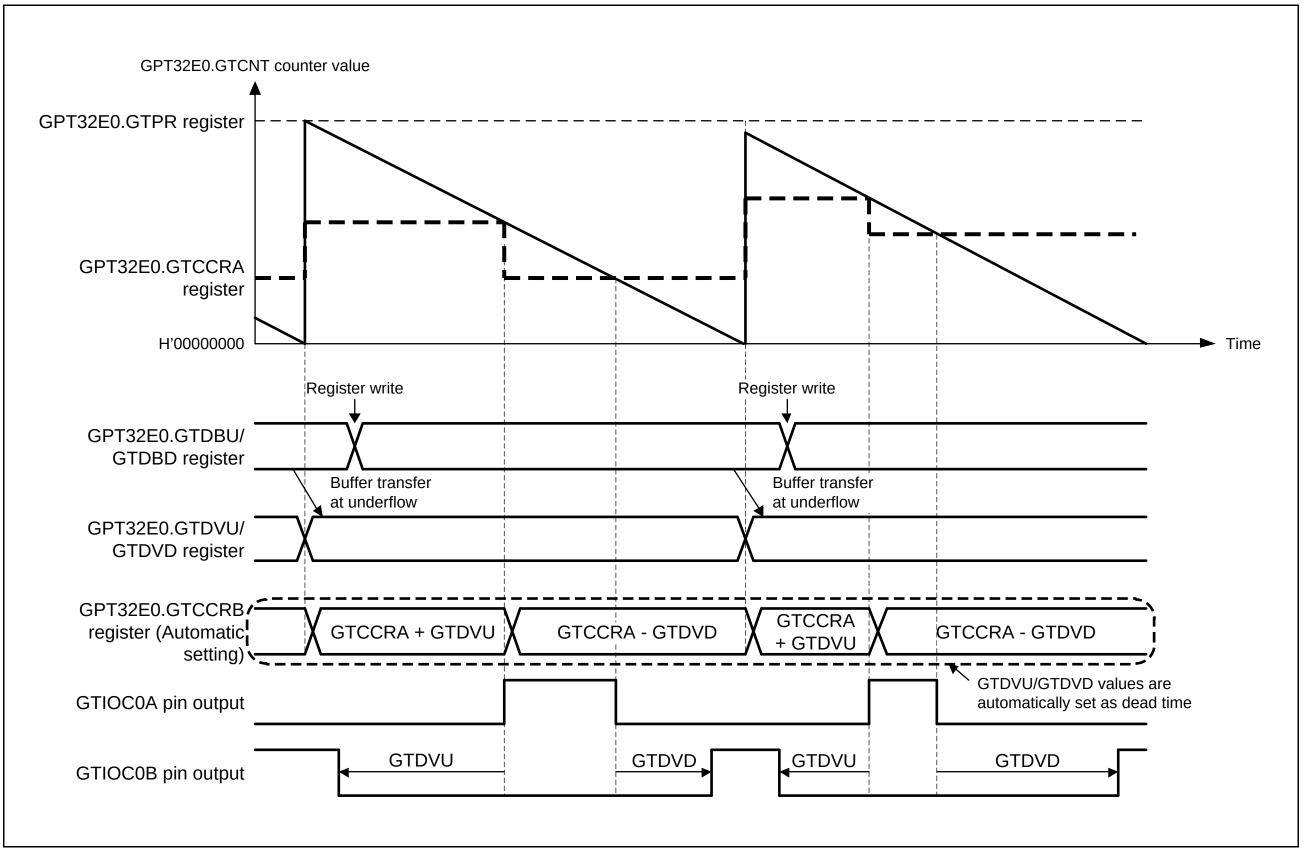


Figure 18.43 Example of automatic dead time setting function operation with saw-wave one-shot pulse mode, down-counting, GTDVU and GTDVD set to buffer operation, and active-high

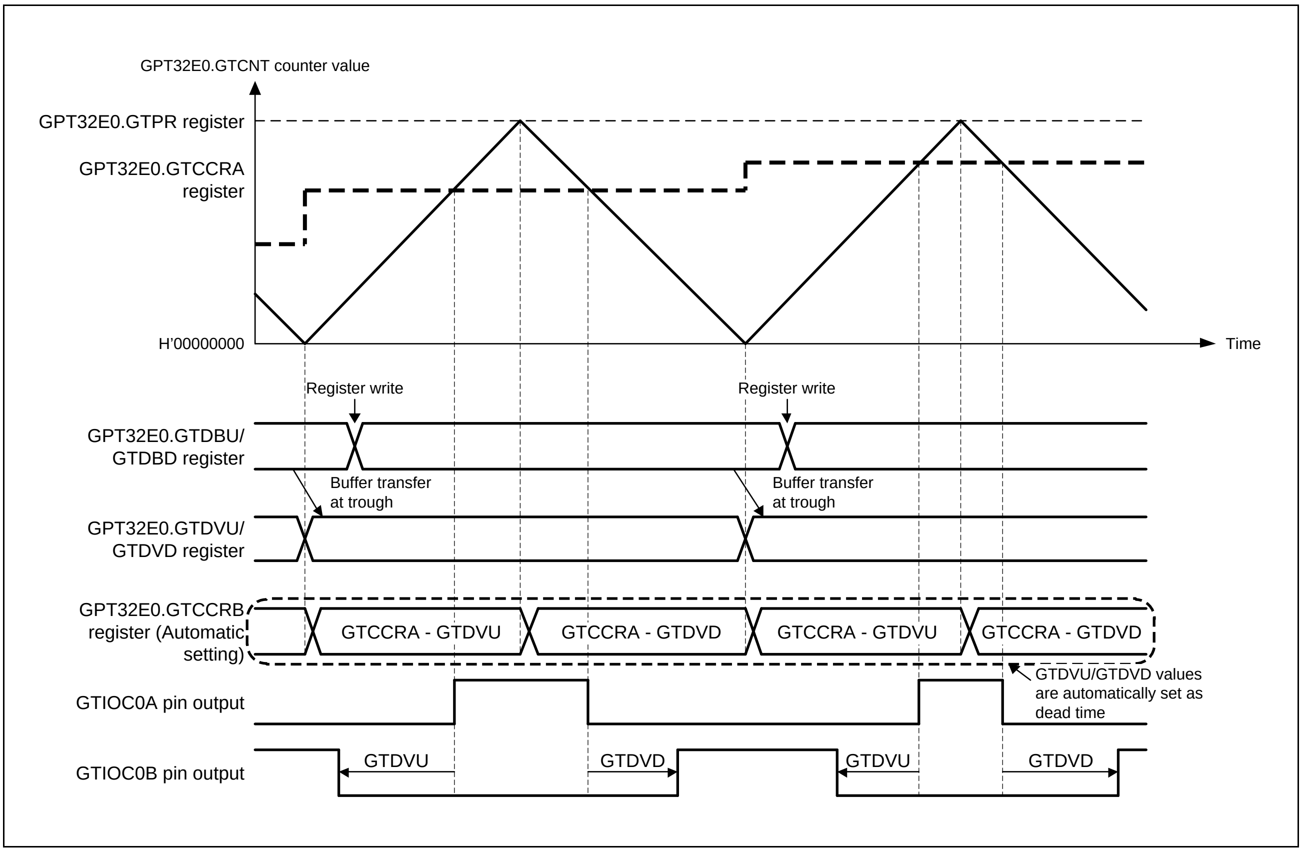


Figure 18.44 Example of automatic compare-match value setting function with dead time with triangle-wave PWM mode 1, GTDVU and GTDVD set to buffer operation, active-high

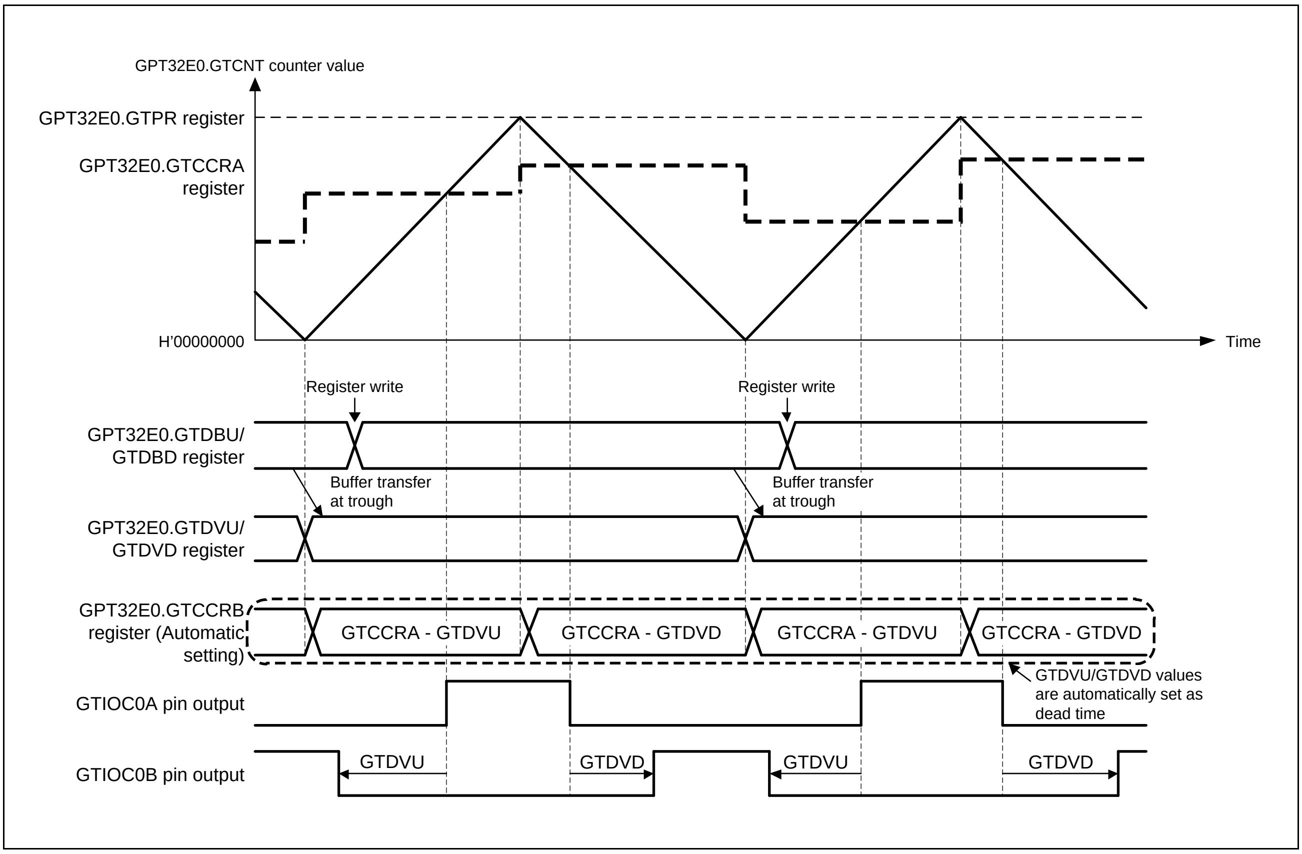
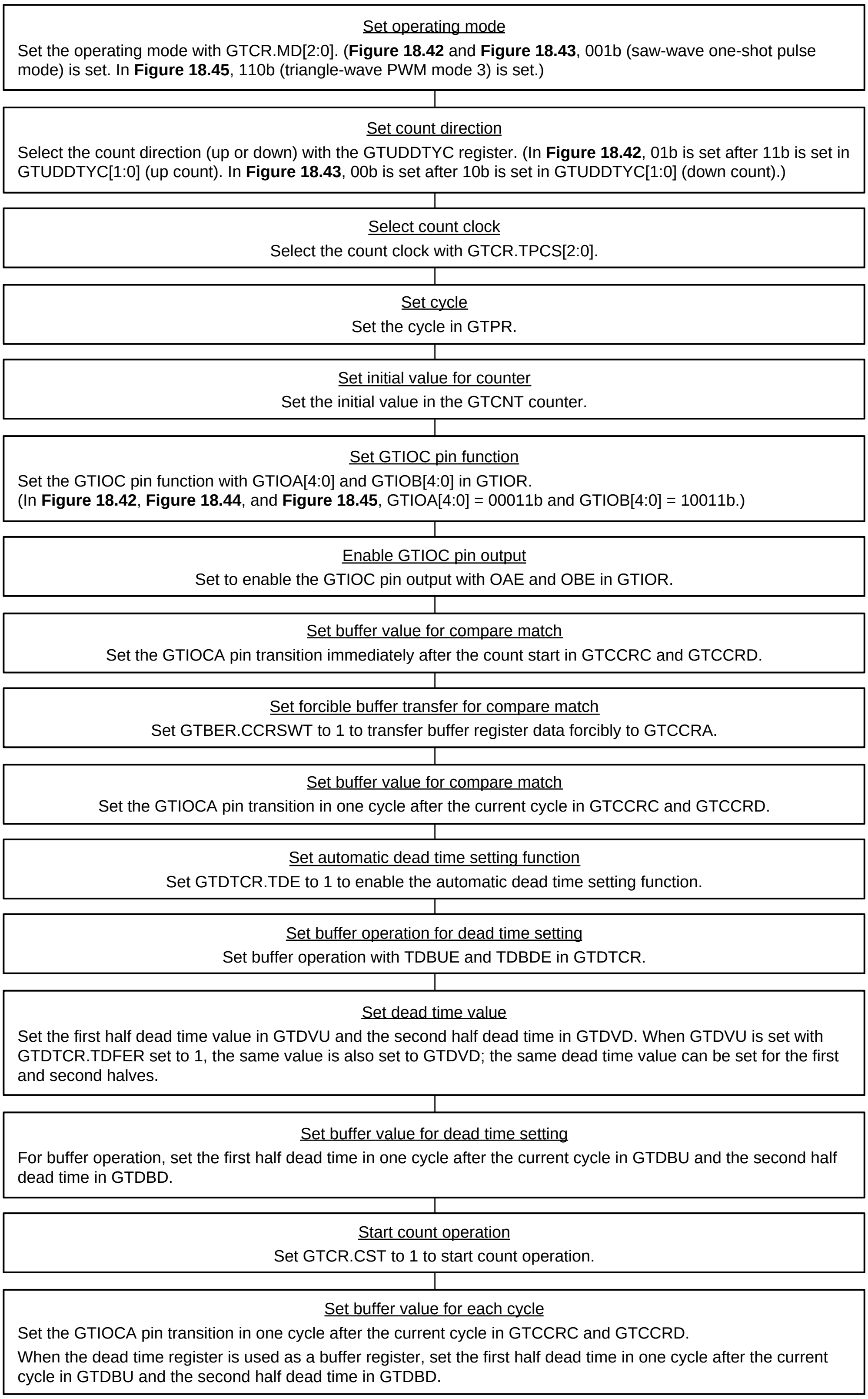


Figure 18.45 Example of automatic compare-match value setting function with dead time, with triangle-wave PWM mode 2 or 3, GTDVU and GTDVD set to buffer operation, and active-high



Set operating mode
Set the operating mode with GTCR.MD[2:0]. (**Figure 18.42** and **Figure 18.43**, 001b (saw-wave one-shot pulse mode) is set. In **Figure 18.45**, 110b (triangle-wave PWM mode 3) is set.)

Set count direction
Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.42**, 01b is set after 11b is set in GTUDDTYC[1:0] (up count). In **Figure 18.43**, 00b is set after 10b is set in GTUDDTYC[1:0] (down count).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.

Set GTIOC pin function
Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR.
(In **Figure 18.42**, **Figure 18.44**, and **Figure 18.45**, GTIOA[4:0] = 00011b and GTIOB[4:0] = 10011b.)

Enable GTIOC pin output
Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer value for compare match
Set the GTIOCA pin transition immediately after the count start in GTCCRC and GTCCRD.

Set forcible buffer transfer for compare match
Set GTBER.CCRSWT to 1 to transfer buffer register data forcibly to GTCCRA.

Set buffer value for compare match
Set the GTIOCA pin transition in one cycle after the current cycle in GTCCRC and GTCCRD.

Set automatic dead time setting function
Set GTDTCR.TDE to 1 to enable the automatic dead time setting function.

Set buffer operation for dead time setting
Set buffer operation with TDBUE and TDBDE in GTDTCR.

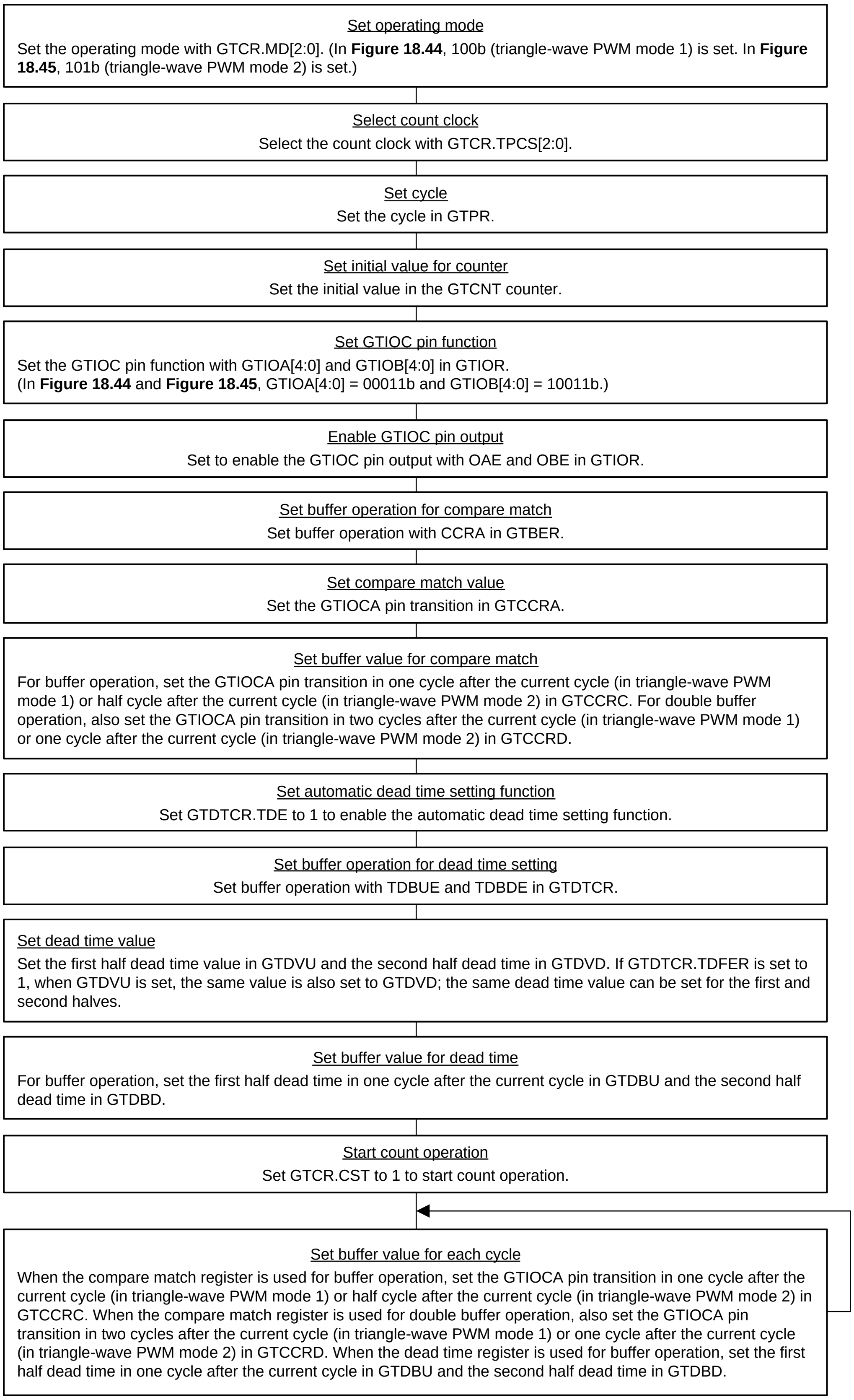
Set dead time value
Set the first half dead time value in GTDVU and the second half dead time in GTDVD. When GTDVU is set with GTDTCR.TDFER set to 1, the same value is also set to GTDVD; the same dead time value can be set for the first and second halves.

Set buffer value for dead time setting
For buffer operation, set the first half dead time in one cycle after the current cycle in GTDBU and the second half dead time in GTDBD.

Start count operation
Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle
Set the GTIOCA pin transition in one cycle after the current cycle in GTCCRC and GTCCRD.
When the dead time register is used as a buffer register, set the first half dead time in one cycle after the current cycle in GTDBU and the second half dead time in GTDBD.

Figure 18.46 Example setting for automatic dead time setting function with saw-wave one-shot pulse mode, and triangle-wave PWM mode 3



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.44**, 100b (triangle-wave PWM mode 1) is set. In **Figure 18.45**, 101b (triangle-wave PWM mode 2) is set.)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter.

Set GTIOC pin function

Set the GTIOC pin function with GTIOA[4:0] and GTIOB[4:0] in GTIOR.
(In **Figure 18.44** and **Figure 18.45**, GTIOA[4:0] = 00011b and GTIOB[4:0] = 10011b.)

Enable GTIOC pin output

Set to enable the GTIOC pin output with OAE and OBE in GTIOR.

Set buffer operation for compare match

Set buffer operation with CCRA in GTBER.

Set compare match value

Set the GTIOCA pin transition in GTCCRA.

Set buffer value for compare match

For buffer operation, set the GTIOCA pin transition in one cycle after the current cycle (in triangle-wave PWM mode 1) or half cycle after the current cycle (in triangle-wave PWM mode 2) in GTCCRC. For double buffer operation, also set the GTIOCA pin transition in two cycles after the current cycle (in triangle-wave PWM mode 1) or one cycle after the current cycle (in triangle-wave PWM mode 2) in GTCCRD.

Set automatic dead time setting function

Set GTDTCR.TDE to 1 to enable the automatic dead time setting function.

Set buffer operation for dead time setting

Set buffer operation with TDBUE and TDBDE in GTDTCR.

Set dead time value

Set the first half dead time value in GTDVU and the second half dead time in GTDVD. If GTDTCR.TDFER is set to 1, when GTDVU is set, the same value is also set to GTDVD; the same dead time value can be set for the first and second halves.

Set buffer value for dead time

For buffer operation, set the first half dead time in one cycle after the current cycle in GTDBU and the second half dead time in GTDBD.

Start count operation

Set GTCR.CST to 1 to start count operation.

Set buffer value for each cycle

When the compare match register is used for buffer operation, set the GTIOCA pin transition in one cycle after the current cycle (in triangle-wave PWM mode 1) or half cycle after the current cycle (in triangle-wave PWM mode 2) in GTCCRC. When the compare match register is used for double buffer operation, also set the GTIOCA pin transition in two cycles after the current cycle (in triangle-wave PWM mode 1) or one cycle after the current cycle (in triangle-wave PWM mode 2) in GTCCRD. When the dead time register is used for buffer operation, set the first half dead time in one cycle after the current cycle in GTDBU and the second half dead time in GTDBD.

Figure 18.47 Example setting for automatic dead time setting function with triangle-wave PWM mode 1 or 2

18.3.5 Count Direction Changing Function

The count direction of the GTCNT counter can be changed by modifying the UD bit in GTUDDTYC.

In saw-wave mode, if the UD bit in GTUDDTYC is modified during count operation, the count direction is changed at an overflow (when modified during up-counting) or an underflow (when modified during down-counting). If the GTUDDTYC.UD bit is modified while the count operation stops and the GTUDDTYC.UDF bit is 0, the GTUDDTYC.UD bit modification is not reflected at the start of counting and the count direction is changed at an overflow or an underflow. If the UDF bit is set to 1 while the count operation stops, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.

In triangle-wave mode, the count direction does not change even though the UD bit in GTUDDTYC is modified during the count operation. Similarly, even though the GTUDDTYC.UD bit is modified while the count operation stops and GTUDDTYC.UDF bit is 0, the GTUDDTYC.UD bit value is not reflected to the count operation. If the GTUDDTYC.UDF bit is set to 1 while the count operation stops, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.

If the count direction changes during a saw-wave count operation, the GTPR value after the start of up-counting is reflected to the count cycle during up-counting and the GTPR value before the start of down-counting is reflected during down-counting.

Figure 18.48 shows an example of count direction changing function operation.

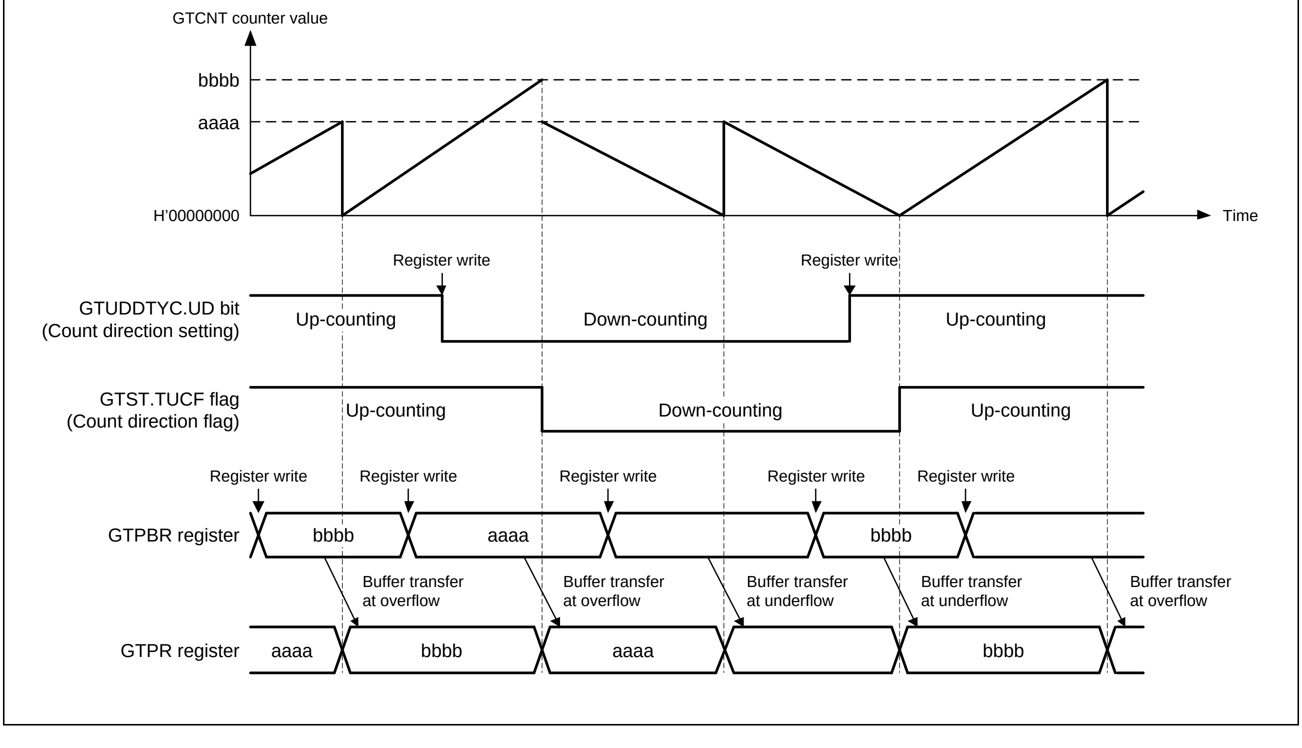


Figure 18.48 Example of count direction changing function operation during buffer operation

18.3.6 Function of Output Duty 0% and 100%

The output duty of the GTIOCA pin and the GTIOCB pin are set to 0% or 100% by changing the GTUDDTYC.OADTY bit or GTUDDTYC.OBDTY bit.

In saw-wave mode, if the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified during the count operation, the output duty setting is reflected at an overflow (when modified during up-counting) or an underflow (when modified during down-counting). If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 0, the output duty modification is not reflected at the start of counting. The output duty changes at an overflow or an underflow. If the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is set to 1 while the count operation stops, the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit value at that time is reflected at the start of counting.

In triangle-wave mode, if the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified during the count operation, the output duty setting is reflected an underflow.

If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 0, the output duty modification is not reflected at the start of counting. The output duty changes at an underflow. If the GTUDDTYC.OADTY bit or the GTUDDTYC.OBDTY bit is modified while the count operation stops and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 1, the output duty modification is reflected at the start of counting.

In performing 0%/100% duty operation, GPT internally continues to:

- Perform compare match operation
- Set compare match flag
- Output interrupt
- Perform buffer operation.

When the control is changed from 0% or 100% duty setting to compare match, the output value of GTIOCA pin at cycle end is decided by GTIOR.GTIOA[3:2] and GTUDDTYC.OADTYR. The output value of GTIOCB pin at cycle end is decided by GTIOR.GTIOB[3:2] and GTUDDTYC.OBDTYR.

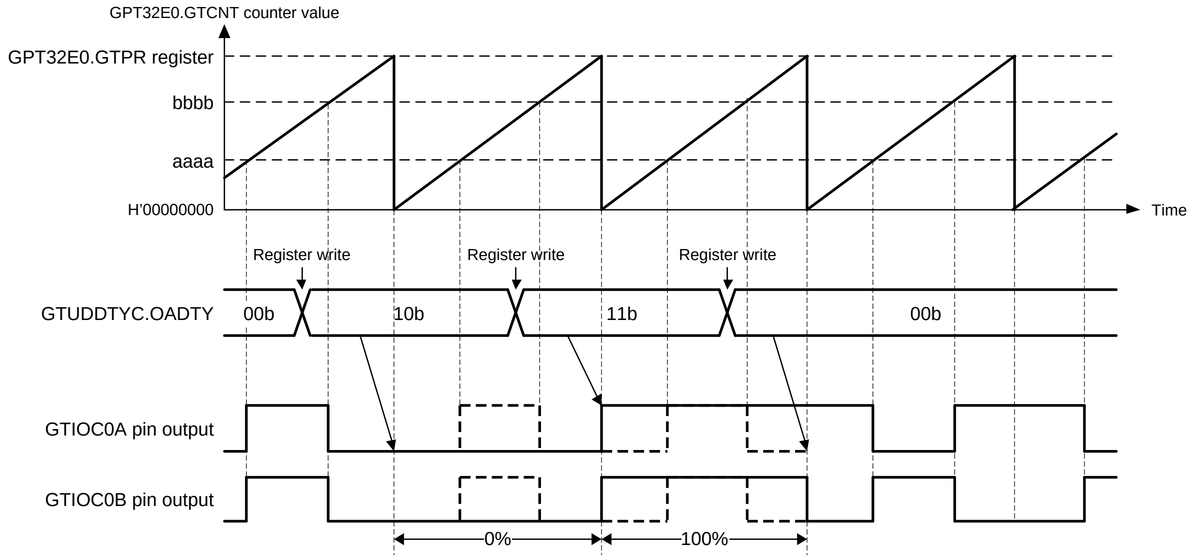
When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 01b, the output pins output low at cycle end. When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 10b, the output pins output high at cycle end.

GTUDDTYC.OADTYR selects the value that is the object of output retained/toggled at cycle end, when GTIOR.GTIOm[3:2] are set to 00b (output retained at cycle end) or when GTIOR.GTIOm[3:2] are set to 11b (output toggled at cycle end). **Table 18.7** shows the values of GTIOCA/GTIOCB pin output at cycle end.

Table 18.7 Output values after releasing 0% or 100% duty setting (m = A, B)

GTIOR.GTIOm[3:2]	Compare match value at cycle end masked by 0% or 100% duty setting	GTUDDTYC.OmDTYR in duty 0% setting		GTUDDTYC.OmDTYR in duty 100% setting	
		0	1	0	1
00 (Output retained at cycle end)	0	0	0	1	0
	1	0	1	1	1
01 (low output at cycle end)	—	0	0	0	0
10 (high output at cycle end)	—	1	1	1	1
11 (Output toggled at cycle end)	0	1	1	0	1
	1	1	0	0	0

Figure 18.49 shows the example of output duty 0% and 100% functions.



[Setting examples]

GPT32E0.GTIOR.GTIOA[4:0] bits:	00011b Initial output of low, output toggled at compare match, output retained at cycle end
GPT32E0.GTUDDTYC.OADTYR bit:	0 Applied the value of duty 0%/100% output to GTIOA[3:2] bits function after 0%/100% duty setting is released.
GPT32E0.GTIOR.GTIOB[4:0] bits:	00011b Initial output of low, output toggled at compare match, output retained at cycle end
GPT32E0.GTUDDTYC.OBDTYR bit:	1 Applied the value of masked compare match output to GTIOB[3:2] bits function after 0%/100% duty setting is released

Figure 18.49 Example of output duty 0% and 100% functions

18.3.7 Hardware Count Start/Count Stop and Clear Operation

The GTCNT counter can be started, stopped, or cleared by the following hardware sources:

- External trigger input
- GTIOCA/GTIOCB pin input.

18.3.7.1 Hardware Start Operation

The GTCNT counter can be started by selecting a hardware source using GTSSR.

Figure 18.50 shows an example of a count start operation by a hardware source. **Figure 18.51** shows the setting example.

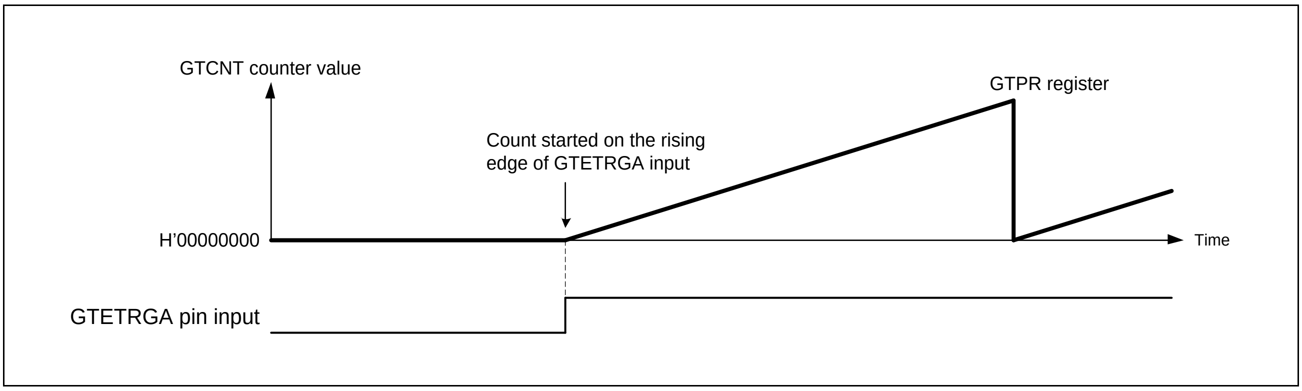
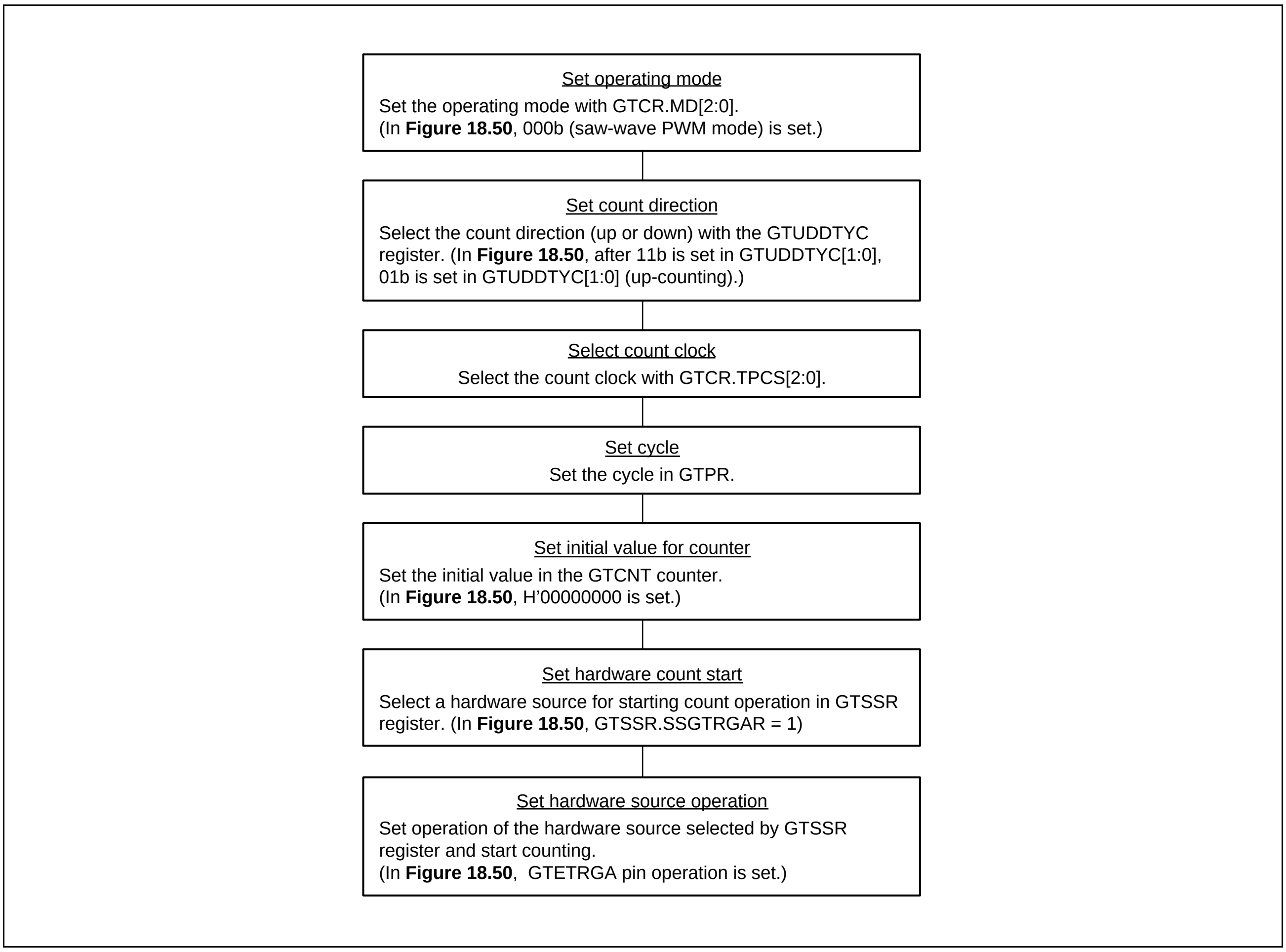


Figure 18.50 Example of count start operation by hardware source, started on the rising edge of GTETRGA pin input



Set operating mode
Set the operating mode with GTCR.MD[2:0].
(In **Figure 18.50**, 000b (saw-wave PWM mode) is set.)

Set count direction
Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.50**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.
(In **Figure 18.50**, H’00000000 is set.)

Set hardware count start
Select a hardware source for starting count operation in GTSSR register. (In **Figure 18.50**, GTSSR.SSGTRGAR = 1)

Set hardware source operation
Set operation of the hardware source selected by GTSSR register and start counting.
(In **Figure 18.50**, GTETRGA pin operation is set.)

Figure 18.51 Example setting for count start operation by hardware source

18.3.7.2 Hardware Stop Operation

The GTCNT counter can be stopped by selecting a hardware source using GTPSR. **Figure 18.52** shows an example of a count stop operation by a hardware source. **Figure 18.53** shows the setting example. In this example, the count operation stops at the rising edge of GTETRGA pin input and restarts at the rising edge of GTETRGB pin input.

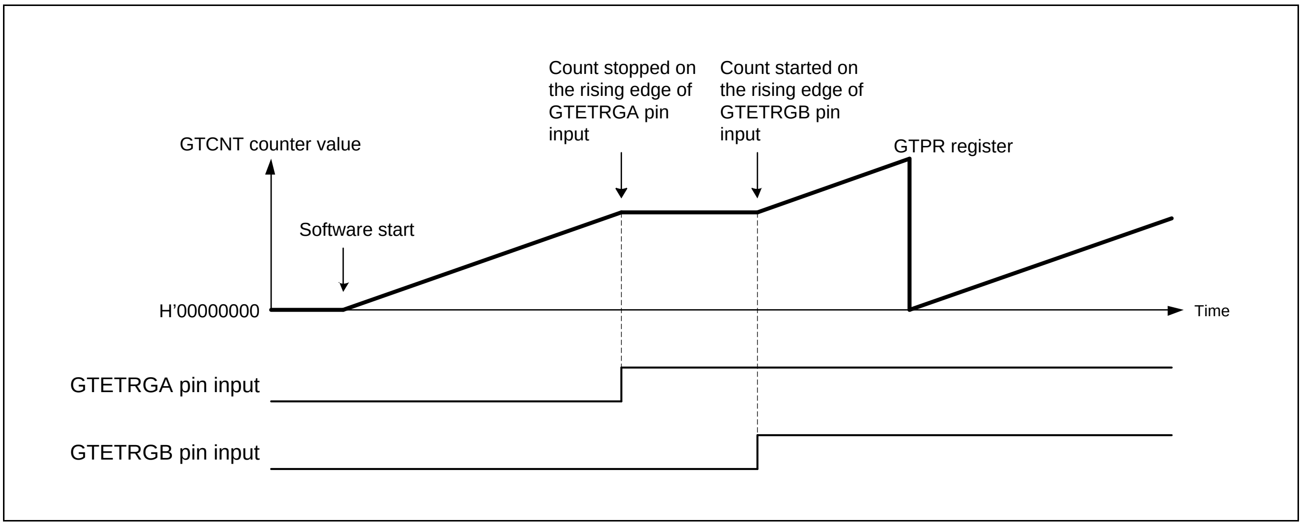
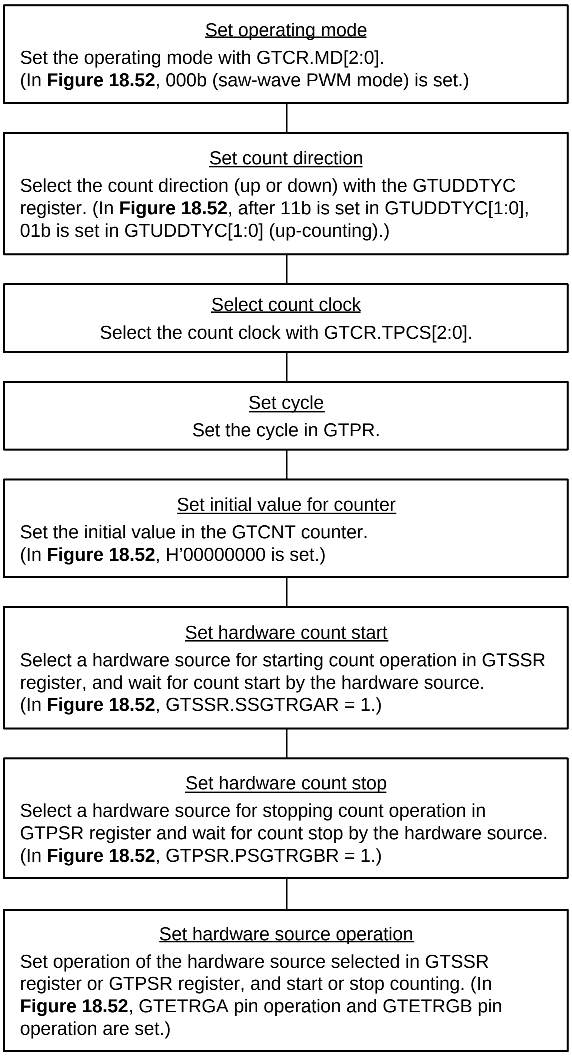


Figure 18.52 Example of count start operation by hardware source, started by software, stopped on the rising edge of GTETRGA pin input, and restarted on the rising edge of GTETRGB pin input



Set operating mode

Set the operating mode with GTCR.MD[2:0]. (In **Figure 18.52**, 000b (saw-wave PWM mode) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.52**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in GTPR.

Set initial value for counter

Set the initial value in the GTCNT counter. (In **Figure 18.52**, H'00000000 is set.)

Set hardware count start

Select a hardware source for starting count operation in GTSSR register, and wait for count start by the hardware source. (In **Figure 18.52**, GTSSR.SSGTRGAR = 1.)

Set hardware count stop

Select a hardware source for stopping count operation in GTPSR register and wait for count stop by the hardware source. (In **Figure 18.52**, GTPSR.PSGTRGBR = 1.)

Set hardware source operation

Set operation of the hardware source selected in GTSSR register or GTPSR register, and start or stop counting. (In **Figure 18.52**, GTETRGA pin operation and GTETRGB pin operation are set.)

Figure 18.53 Example setting for count stop operation by hardware source

Figure 18.54 shows an example of a count start/stop operation by a hardware source. **Figure 18.55** shows the setting example. In this example, the counter operates during the high-level periods of the external trigger input GTETRGA.

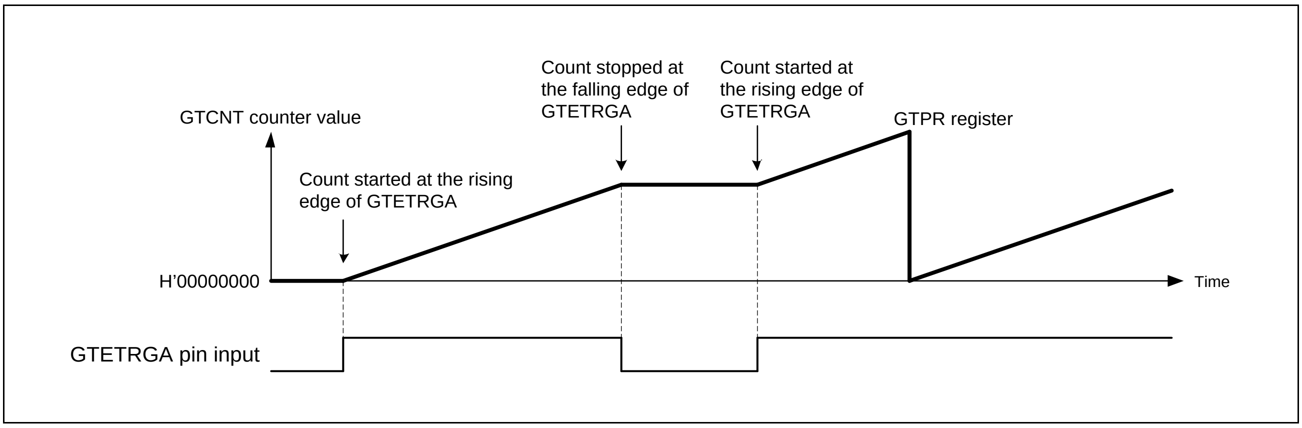
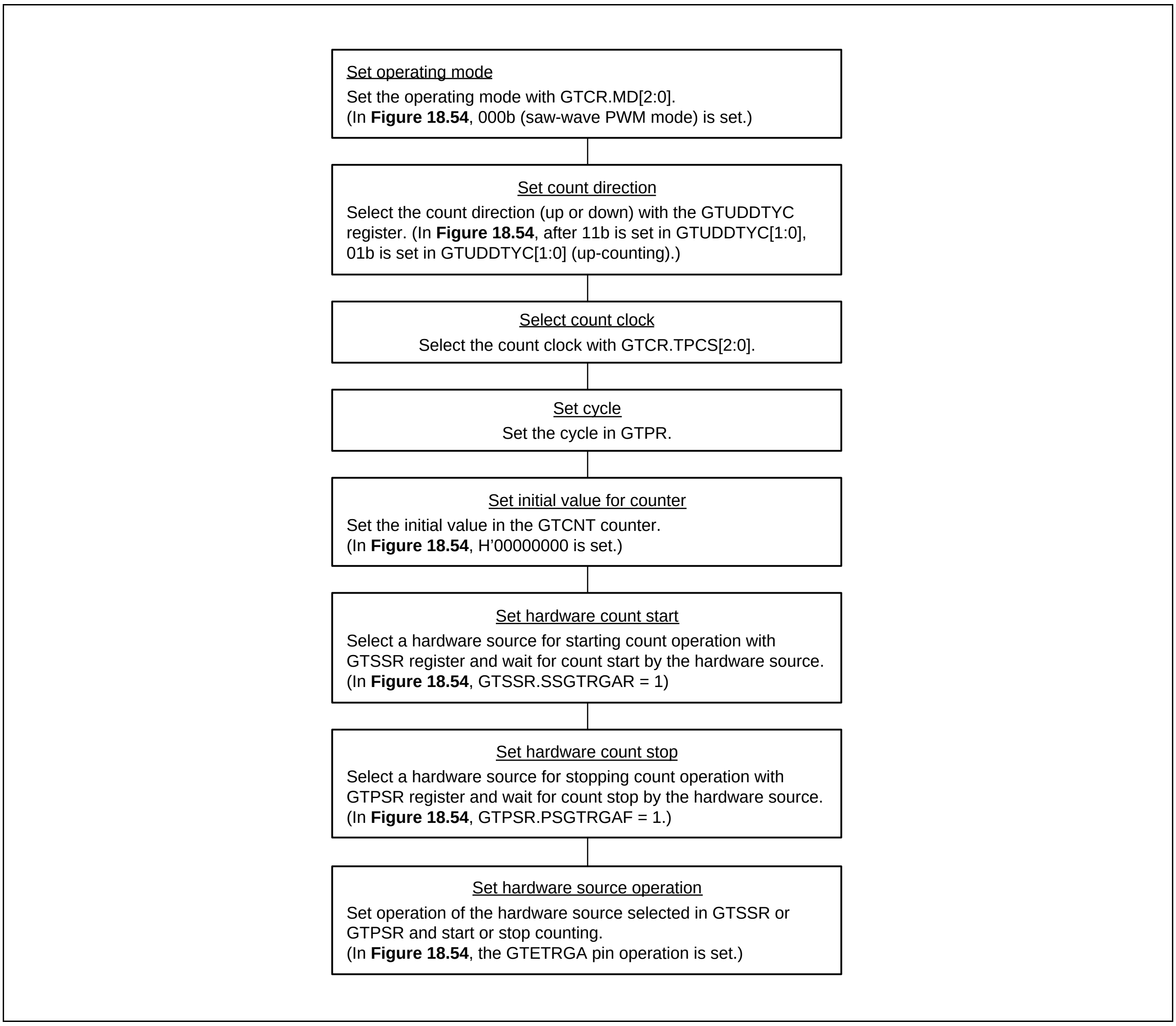


Figure 18.54 Example of count start/stop operation by hardware source, started on the rising edge of the GTETRGA pin input and stopped on the falling edge of the GTETRGA pin input



Set operating mode
Set the operating mode with GTCR.MD[2:0].
(In **Figure 18.54**, 000b (saw-wave PWM mode) is set.)

Set count direction
Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.54**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in GTPR.

Set initial value for counter
Set the initial value in the GTCNT counter.
(In **Figure 18.54**, H'00000000 is set.)

Set hardware count start
Select a hardware source for starting count operation with GTSSR register and wait for count start by the hardware source.
(In **Figure 18.54**, GTSSR.SSGTRGAR = 1)

Set hardware count stop
Select a hardware source for stopping count operation with GTPSR register and wait for count stop by the hardware source.
(In **Figure 18.54**, GTPSR.PSGTRGAF = 1.)

Set hardware source operation
Set operation of the hardware source selected in GTSSR or GTPSR and start or stop counting.
(In **Figure 18.54**, the GTETRGA pin operation is set.)

Figure 18.55 Example setting for count start/stop operation by hardware source

18.3.7.3 Hardware Clear Operation

The GTCNT counter can be cleared by selecting a hardware source using GTCSR. The OVFn/UNFn (n=0 to 7) interrupt (overflow/underflow interrupt) is not generated when the GTCNT counter is cleared by a hardware source or by software.

Figure 18.56 and **Figure 18.57** show examples of the GTCNT counter clearing operation by a hardware source. **Figure 18.58** shows the setting example. In this example, the GTCNT counter starts at the rising edge of GTETRGA pin input, and the counter stops/clears at the falling edge of GTETRGA pin input.

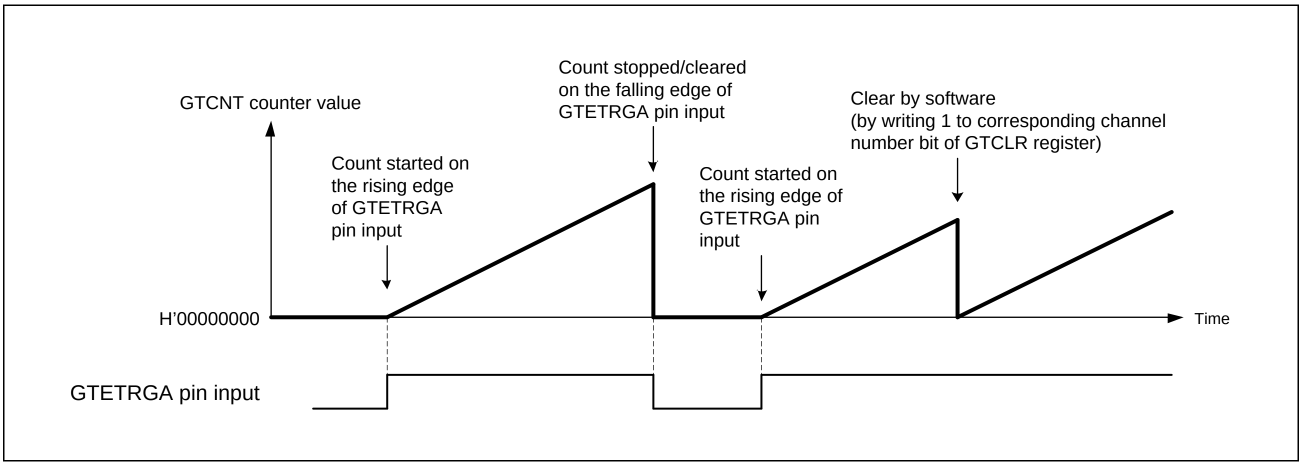


Figure 18.56 Example of count clearing operation by hardware source with saw wave up-counting, started on the rising edge of GTETRGA pin input, and stopped/cleared on the falling edge of GTETRGA pin input

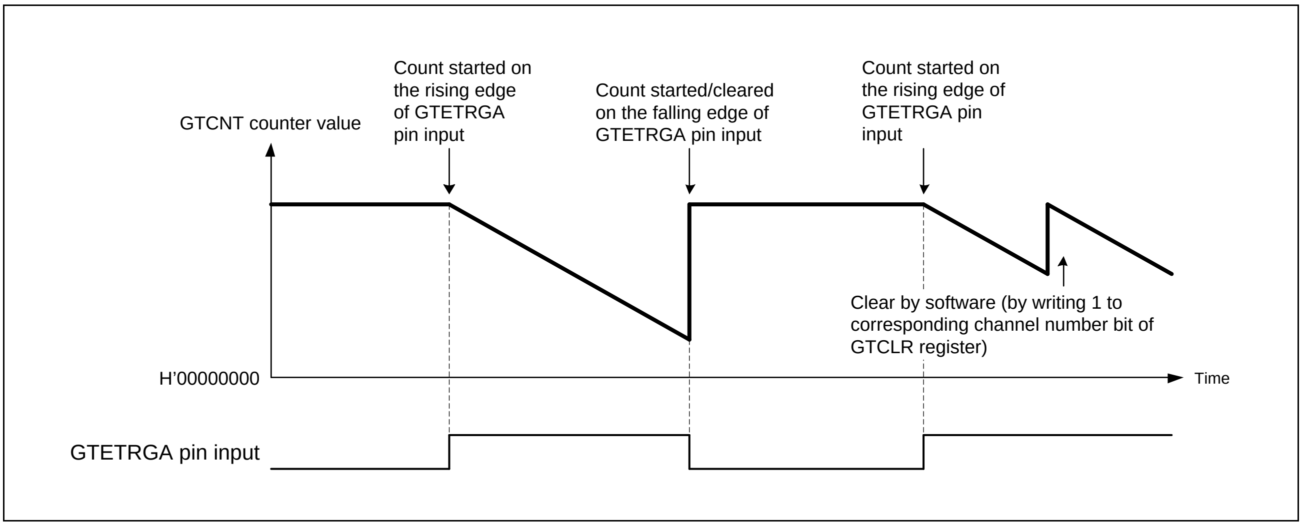
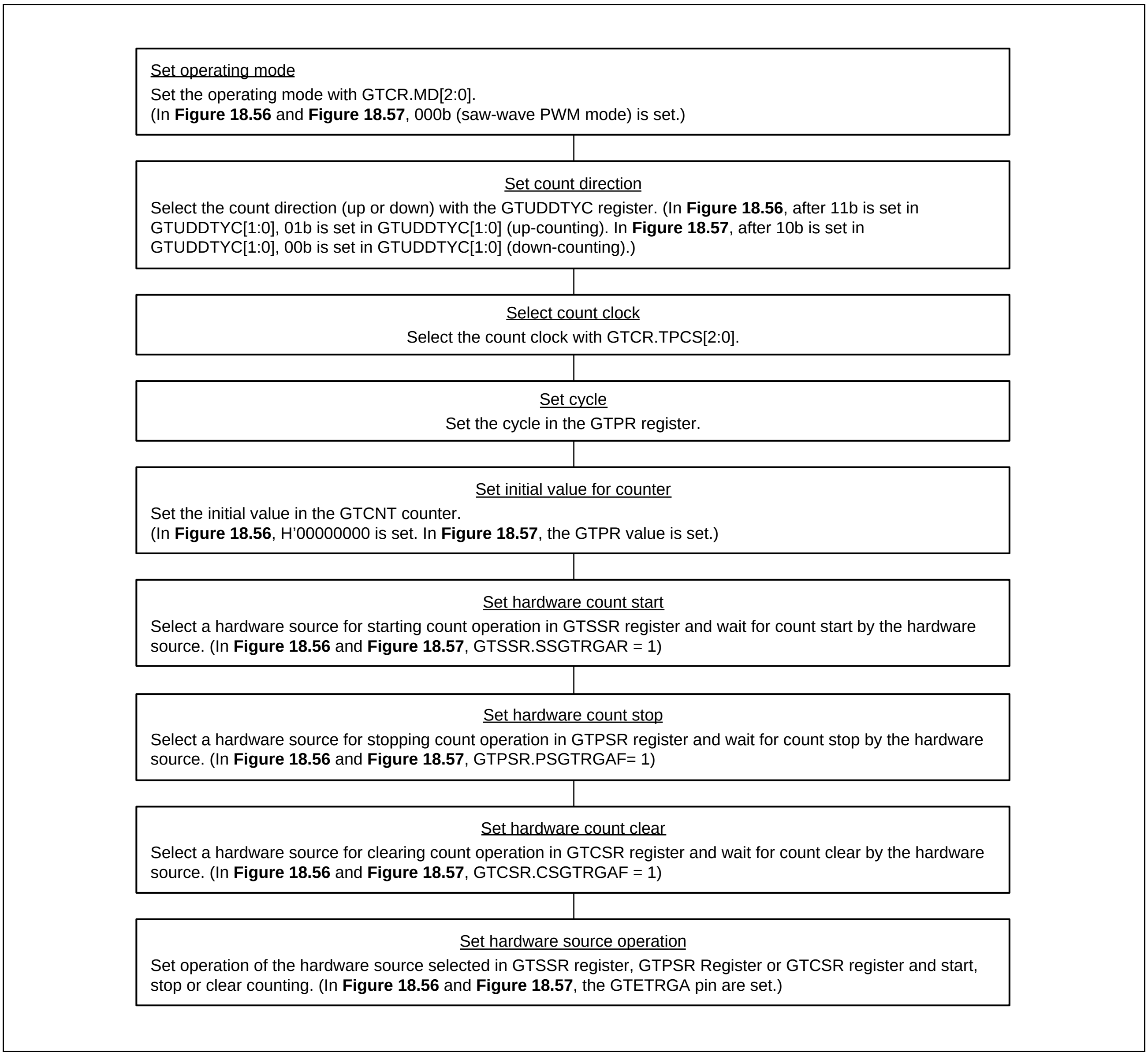


Figure 18.57 Examples of count clearing operation by hardware source with saw wave down-counting, started at event input A, and stopped/cleared at event input B



Set operating mode
Set the operating mode with GTCR.MD[2:0].
(In **Figure 18.56** and **Figure 18.57**, 000b (saw-wave PWM mode) is set.)

Set count direction
Select the count direction (up or down) with the GTUDDTYC register. (In **Figure 18.56**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting). In **Figure 18.57**, after 10b is set in GTUDDTYC[1:0], 00b is set in GTUDDTYC[1:0] (down-counting).)

Select count clock
Select the count clock with GTCR.TPCS[2:0].

Set cycle
Set the cycle in the GTPR register.

Set initial value for counter
Set the initial value in the GTCNT counter.
(In **Figure 18.56**, H'00000000 is set. In **Figure 18.57**, the GTPR value is set.)

Set hardware count start
Select a hardware source for starting count operation in GTSSR register and wait for count start by the hardware source. (In **Figure 18.56** and **Figure 18.57**, GTSSR.SSGTRGAR = 1)

Set hardware count stop
Select a hardware source for stopping count operation in GTPSR register and wait for count stop by the hardware source. (In **Figure 18.56** and **Figure 18.57**, GTPSR.PSGTRGAF= 1)

Set hardware count clear
Select a hardware source for clearing count operation in GTCSR register and wait for count clear by the hardware source. (In **Figure 18.56** and **Figure 18.57**, GTCSR.CSGTRGAF = 1)

Set hardware source operation
Set operation of the hardware source selected in GTSSR register, GTPSR Register or GTCSR register and start, stop or clear counting. (In **Figure 18.56** and **Figure 18.57**, the GTETRGA pin are set.)

Figure 18.58 Example setting for count clearing operation by hardware source

The OVFn/UNFn (n=0 to 7) interrupt (overflow/underflow interrupt) is not generated when the counter is cleared by a hardware source or by software.

Figure 18.59 shows the relationship between the counter clearing by a hardware source and the OVFn (n=0 to 7) interrupt.

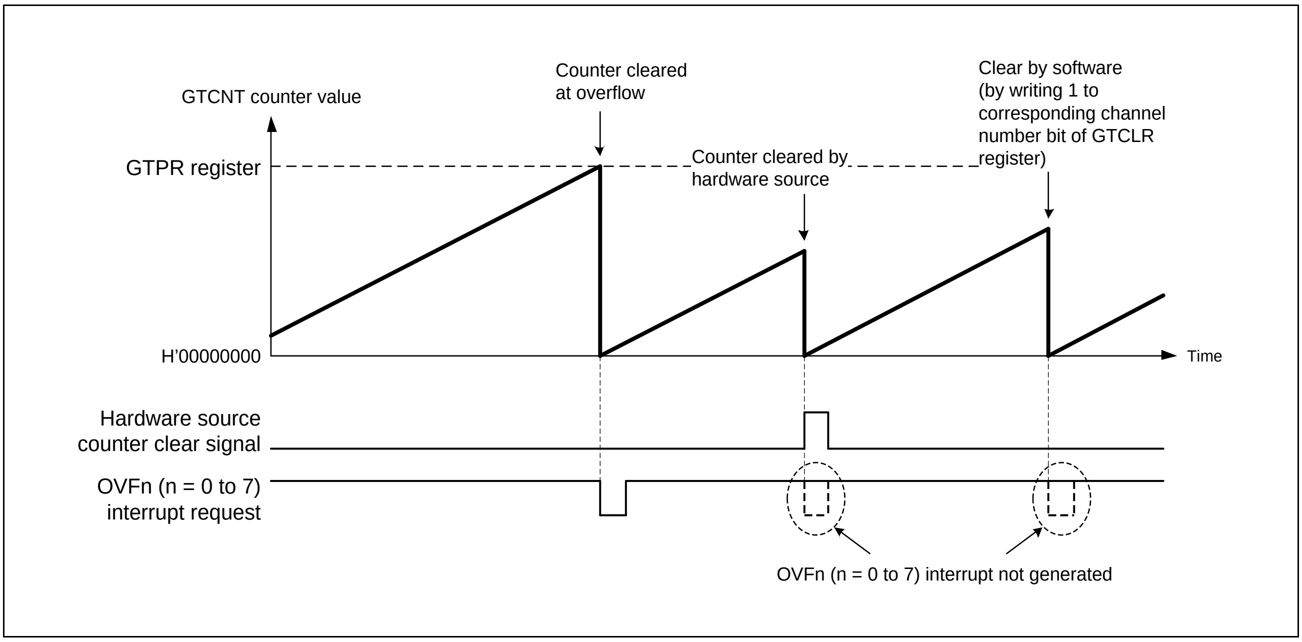


Figure 18.59 Relationship between counter clearing by hardware source and OVFn (n = 0 to 7) interrupt

18.3.8 Synchronized Operation

Synchronized operation on channels such as a synchronized start, stop and clear operation can be performed.

18.3.8.1 Synchronized Operation by Software

The GTCNT counters can be started, stopped, and cleared on multiple channels by setting the associated GTSTR, GTSTP or GTCLR bits simultaneously to 1.

Count start with a phase difference is possible by setting the initial value in the GTCNT counter and setting the associated GTSTR bits simultaneously to 1.

Figure 18.60 shows an example of a simultaneous start, stop and clear by software. **Figure 18.61** shows an example of phase start operation by software.

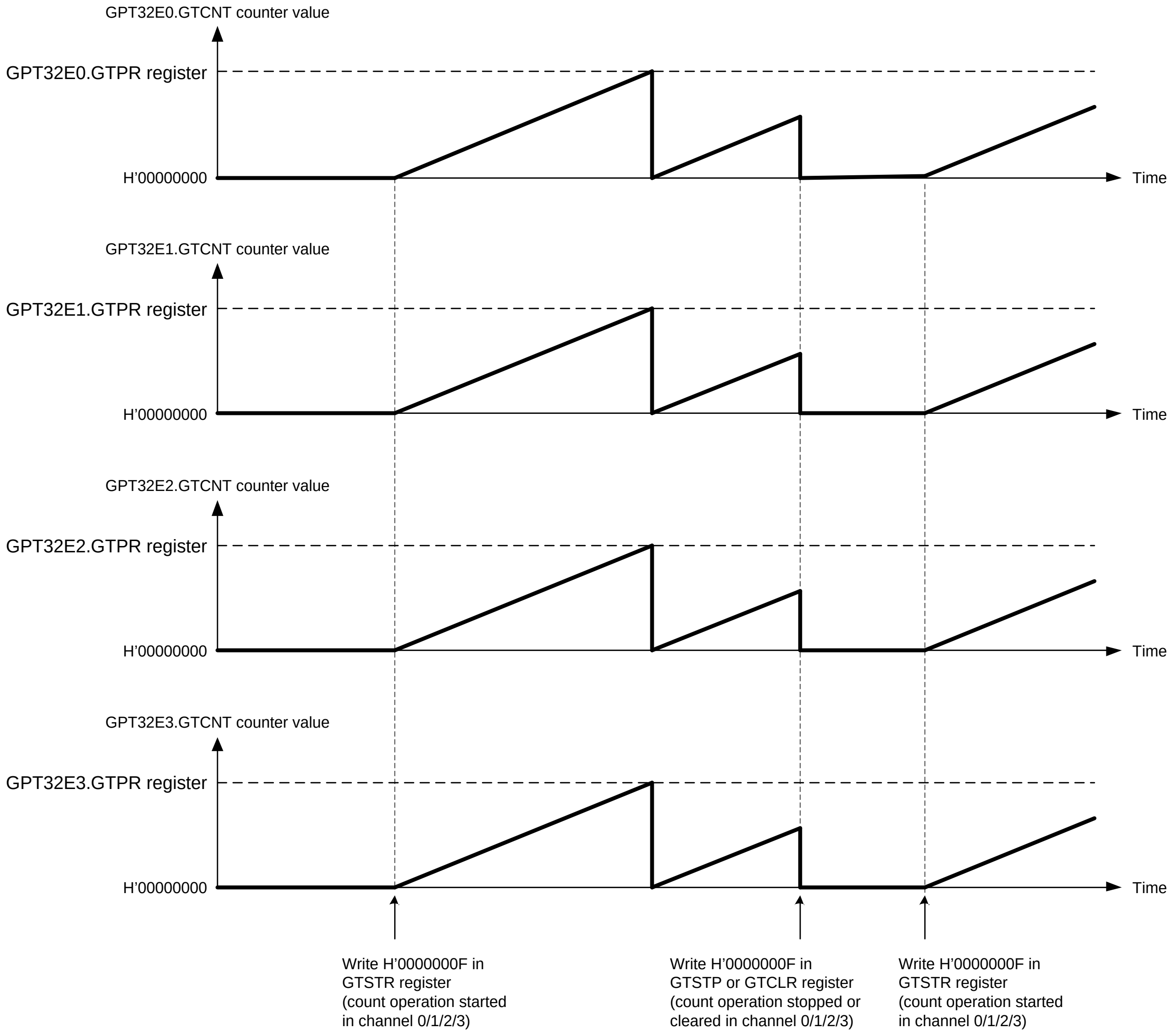


Figure 18.60 Example of a simultaneous start, stop, and clear by software, with the same count cycle (GTPR register value)

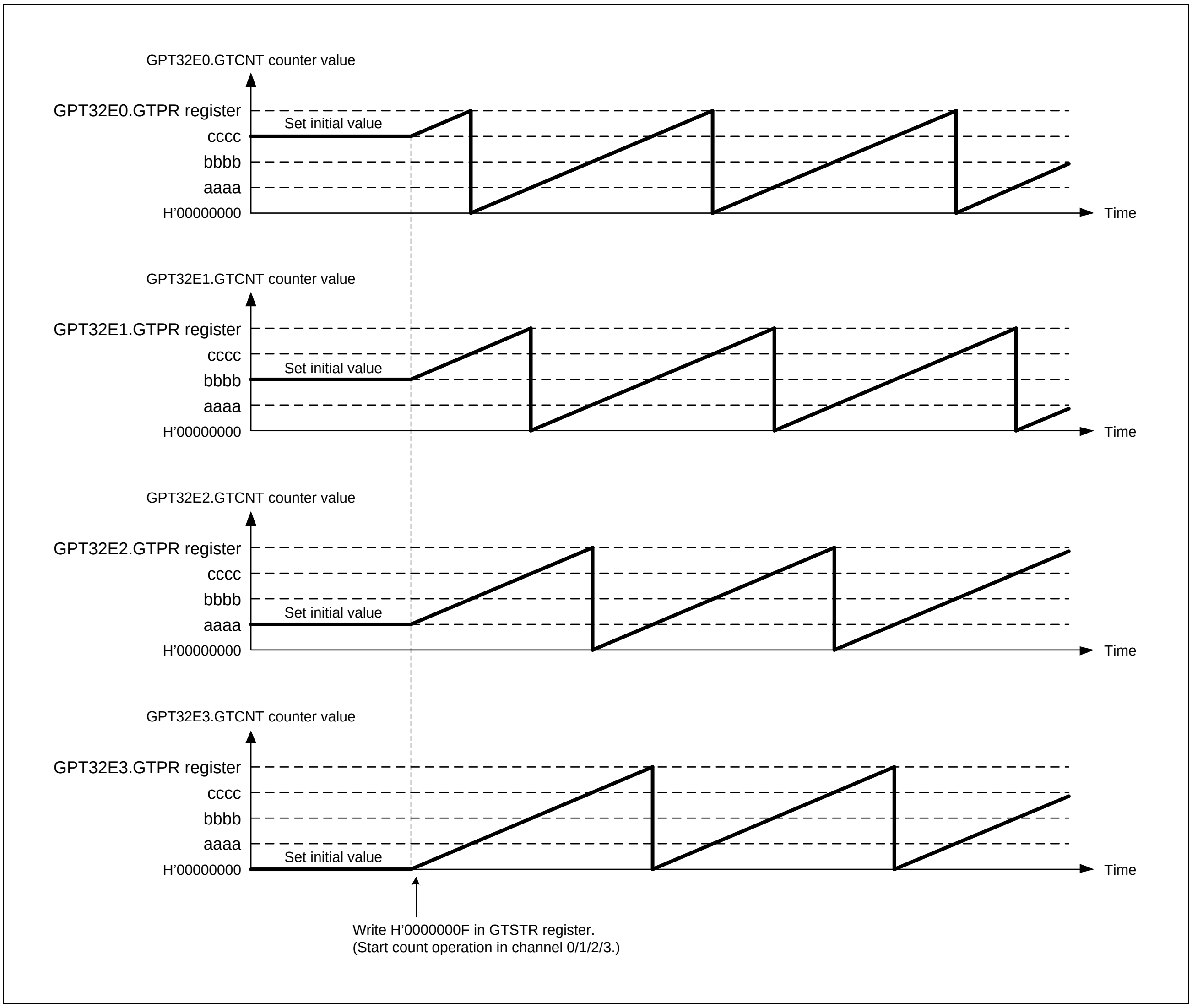


Figure 18.61 Example of software phase start with the same count cycle (GTPR register value)

18.3.8.2 Synchronized Operation by Hardware

The GTCNT counters can be started simultaneously by the following hardware sources:

- External trigger input

Figure 18.62 shows an example of a simultaneous start, stop and clear operation by a hardware source. **Figure 18.63** shows the setting example.

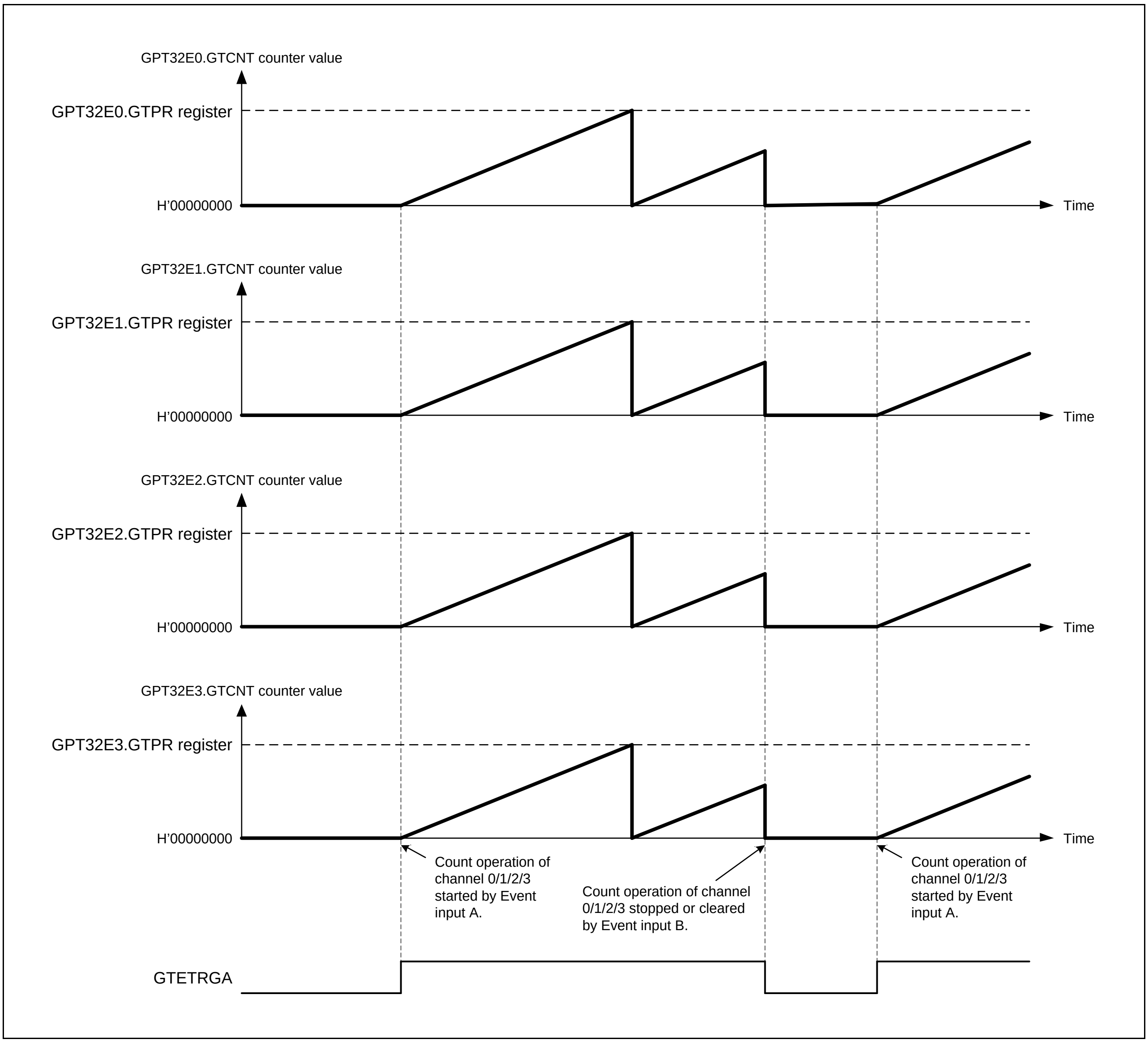
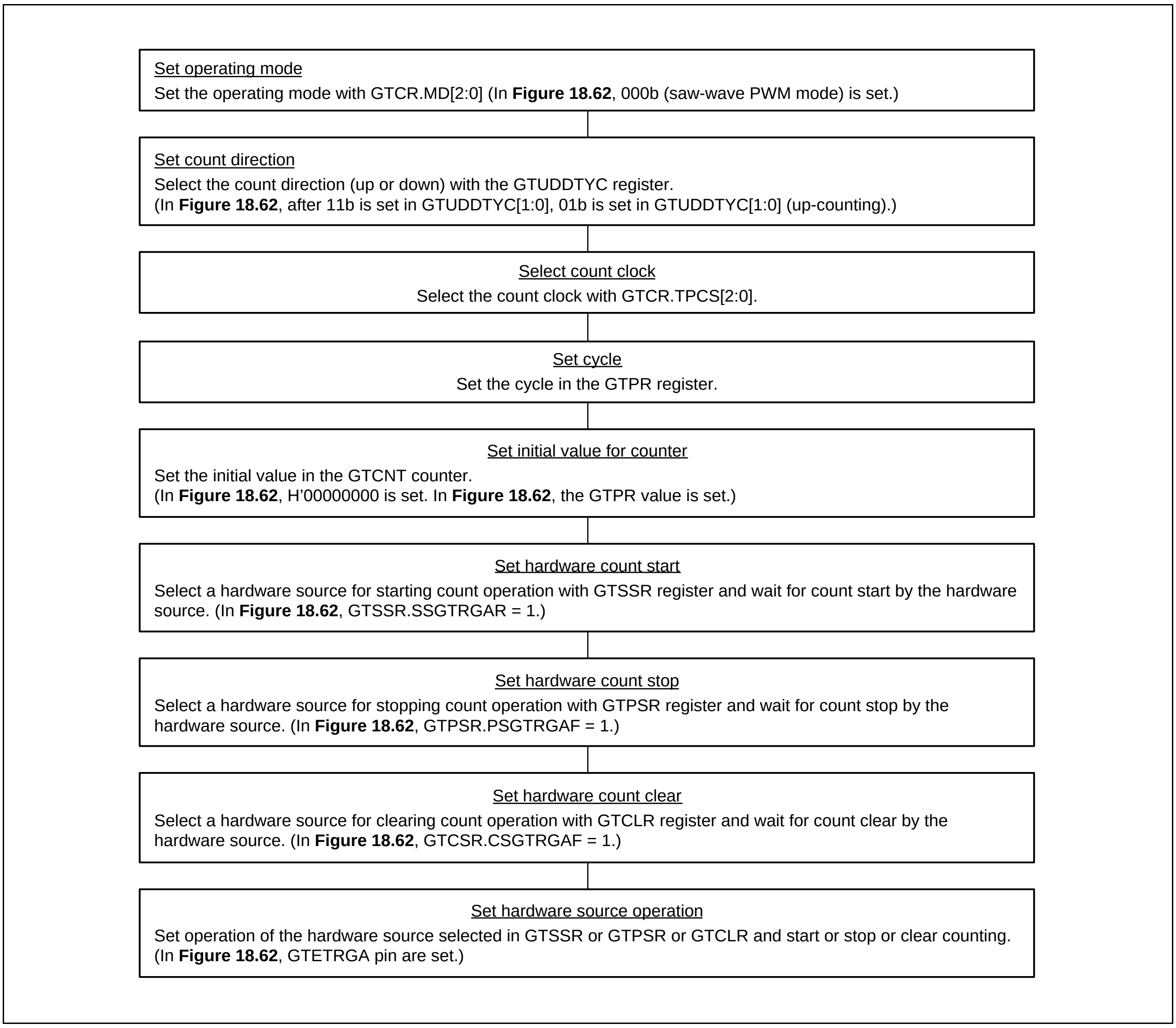


Figure 18.62 Example of a simultaneous start, stop, and clear by the hardware sources, with the same count cycle (GTPR register value)



Set operating mode

Set the operating mode with GTCR.MD[2:0] (In **Figure 18.62**, 000b (saw-wave PWM mode) is set.)

Set count direction

Select the count direction (up or down) with the GTUDDTYC register.
(In **Figure 18.62**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).)

Select count clock

Select the count clock with GTCR.TPCS[2:0].

Set cycle

Set the cycle in the GTPR register.

Set initial value for counter

Set the initial value in the GTCNT counter.
(In **Figure 18.62**, H'00000000 is set. In **Figure 18.62**, the GTPR value is set.)

Set hardware count start

Select a hardware source for starting count operation with GTSSR register and wait for count start by the hardware source. (In **Figure 18.62**, GTSSR.SSGTRGAR = 1.)

Set hardware count stop

Select a hardware source for stopping count operation with GTPSR register and wait for count stop by the hardware source. (In **Figure 18.62**, GTPSR.PSGTRGAF = 1.)

Set hardware count clear

Select a hardware source for clearing count operation with GTCLR register and wait for count clear by the hardware source. (In **Figure 18.62**, GTCSR.CSGTRGAF = 1.)

Set hardware source operation

Set operation of the hardware source selected in GTSSR or GTPSR or GTCLR and start or stop or clear counting.
(In **Figure 18.62**, GTETRGA pin are set.)

Figure 18.63 Example setting for simultaneous start by hardware source

18.3.9 PWM Output Operation Examples

18.3.9.1 Synchronized PWM Output

The GPT output 16 phases of linked PWM waveforms for a maximum of 8 channels by synchronizing operation on channels.

Figure 18.64 shows an example in which four channels perform synchronized operation in saw-wave PWM mode and eight phases of PWM waveforms are output. The GTIOCA is set so that it outputs low as the initial value, high at a GTCCRA compare match, and low at the cycle end. The GTIOCB is set so that it outputs low as the initial value, high at a GTCCRB compare match, and low at the cycle end.

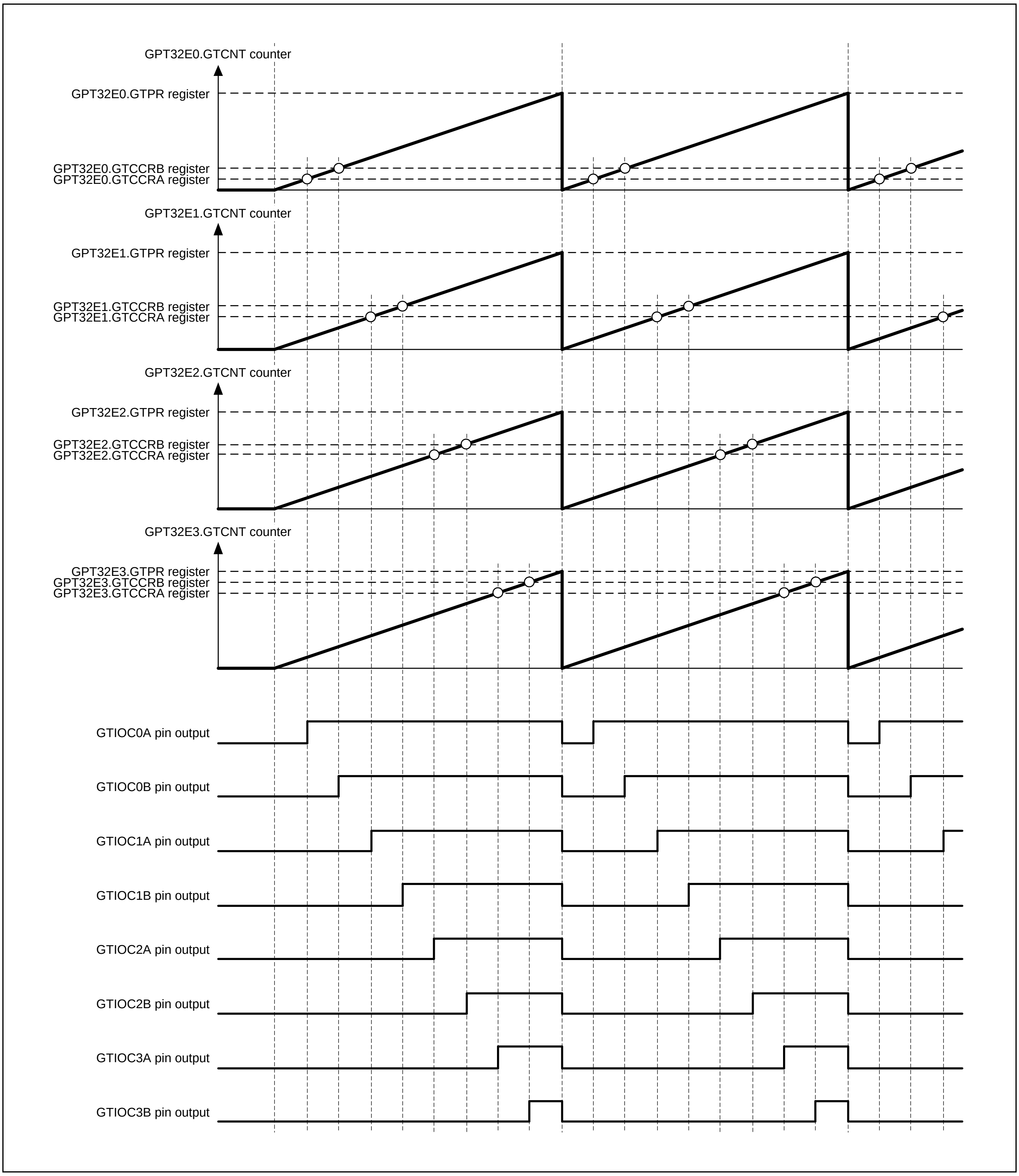


Figure 18.64 Example of synchronized PWM output

18.3.9.2 Three-Phase Saw-Wave Complementary PWM Output

Figure 18.65 shows an example in which three channels perform synchronized operation in saw-wave PWM mode and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, high at a GTCCRA compare match, and low at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, low at a GTCCRB compare match, and high at the cycle end.

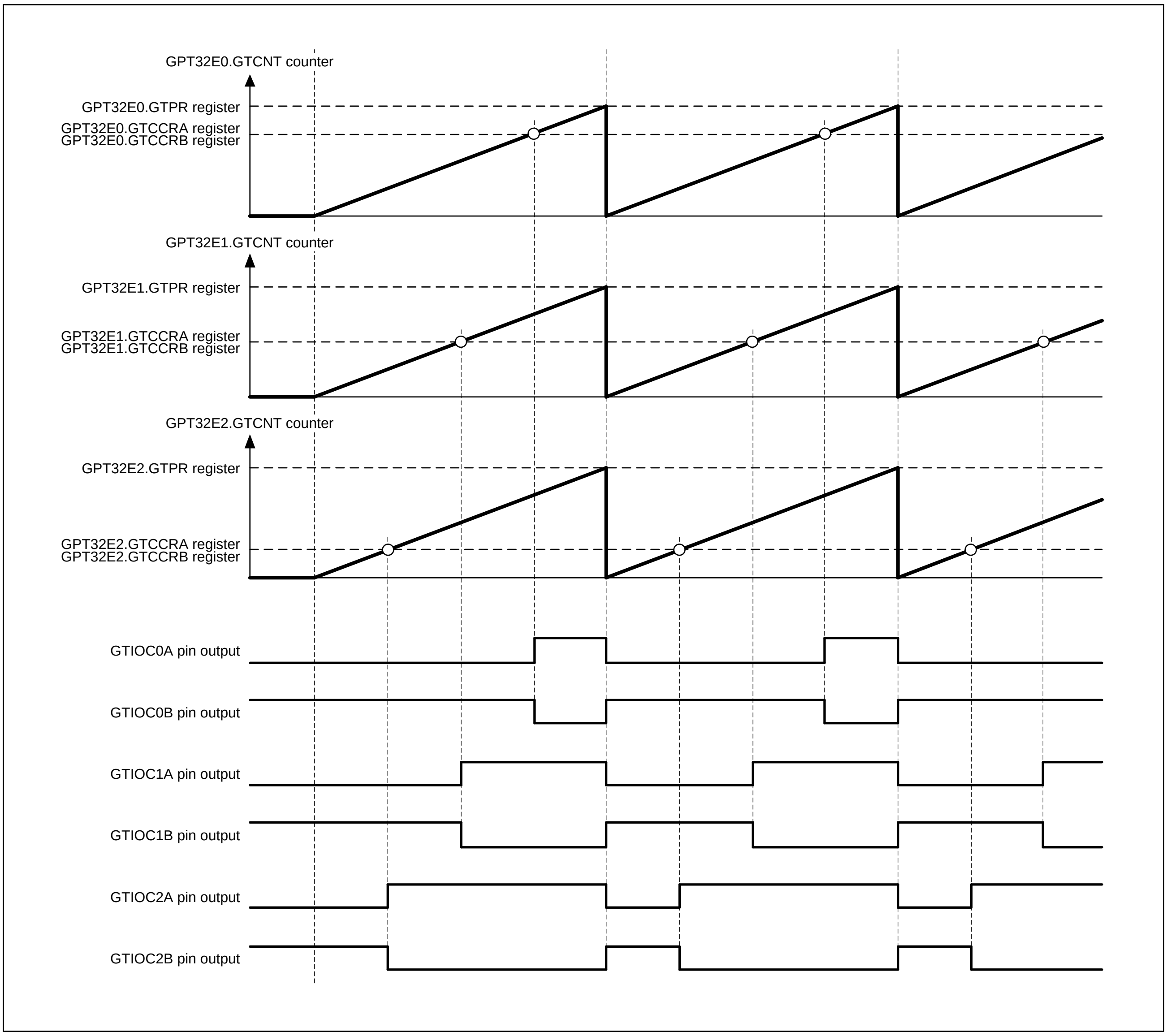


Figure 18.65 Example of 3-phase saw-wave complementary PWM output

18.3.9.3 3-Phase Saw-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 18.66 shows an example in which three channels perform synchronized operation in saw-wave one-shot pulse mode with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggle the output at a GTCCRA compare match, and retain the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggle the output at a GTCCRB compare match, and retain the output at the cycle end.

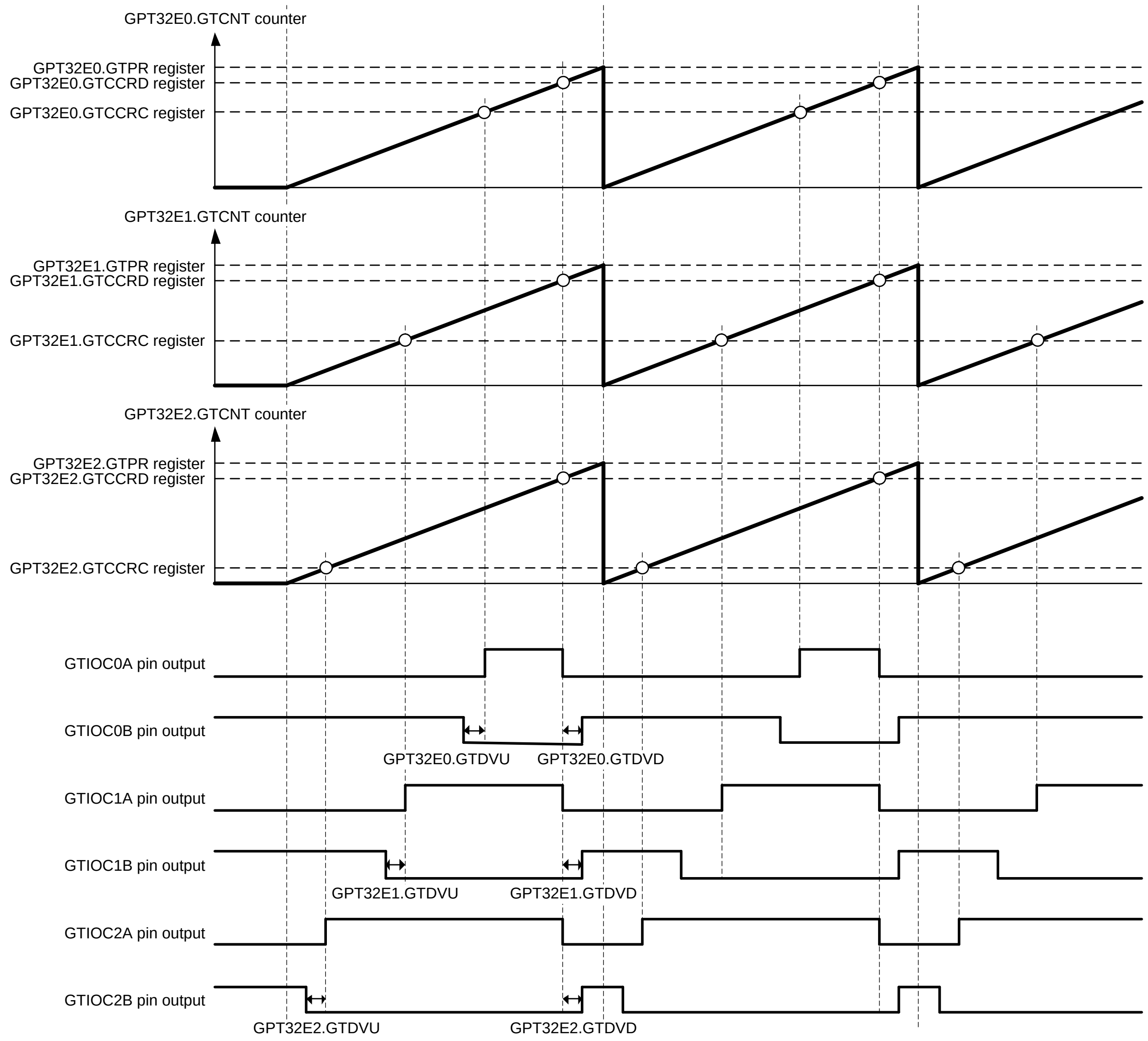


Figure 18.66 Example of 3-phase saw-wave complementary PWM output with automatic dead time setting

18.3.9.4 3-Phase Triangle-Wave Complementary PWM Output

Figure 18.67 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

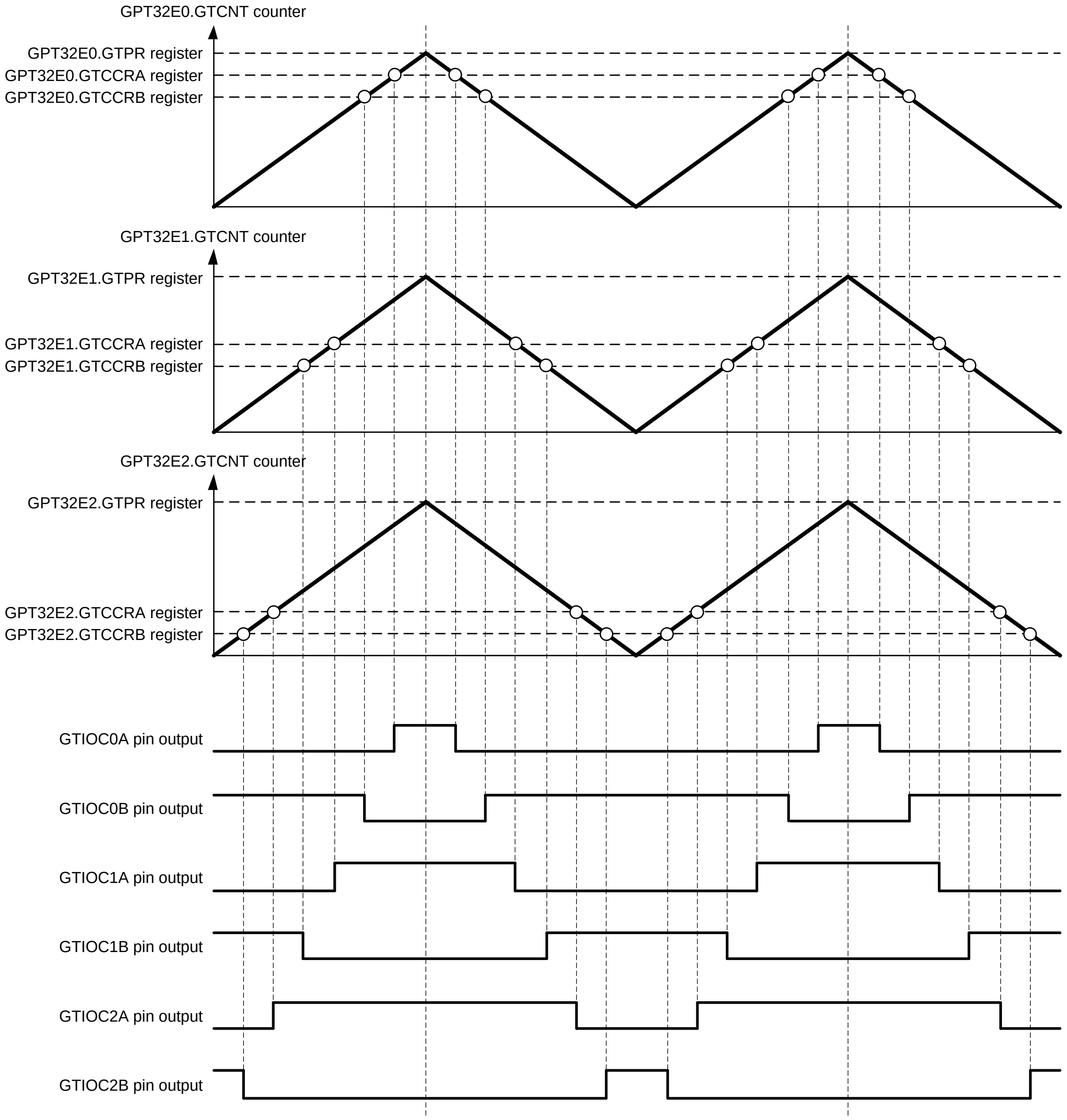


Figure 18.67 Example of 3-phase triangle-wave complementary PWM output

18.3.9.5 3-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 18.68 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA pin is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB pin is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

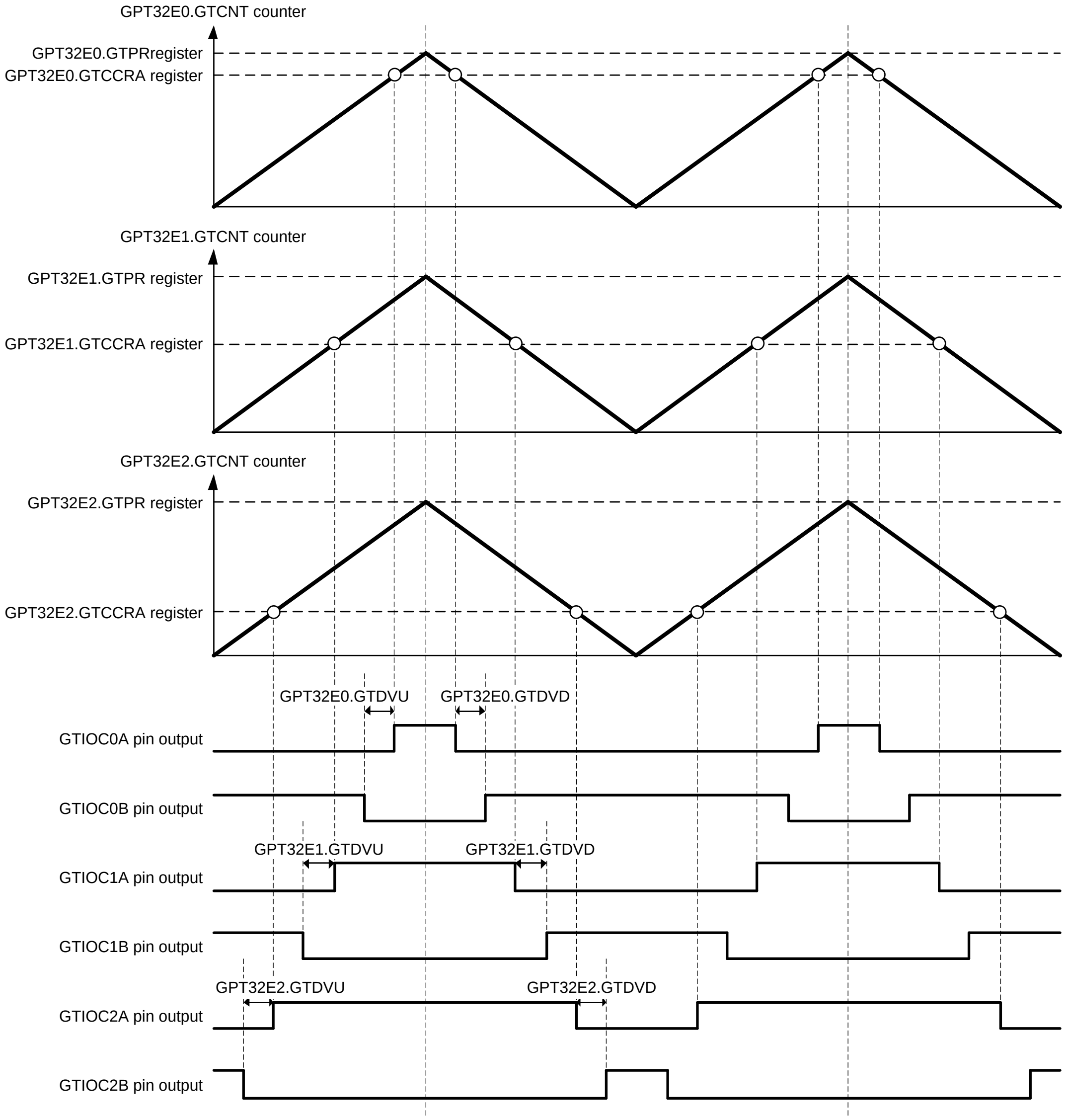


Figure 18.68 Example of 3-phase triangle-wave complementary PWM output with automatic dead time setting

18.3.9.6 3-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 18.69 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 3 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCA is set so that it outputs low as the initial value, toggles the output at a GTCCRA compare match, and retains the output at the cycle end. The GTIOCB is set so that it outputs high as the initial value, toggles the output at a GTCCRB compare match, and retains the output at the cycle end.

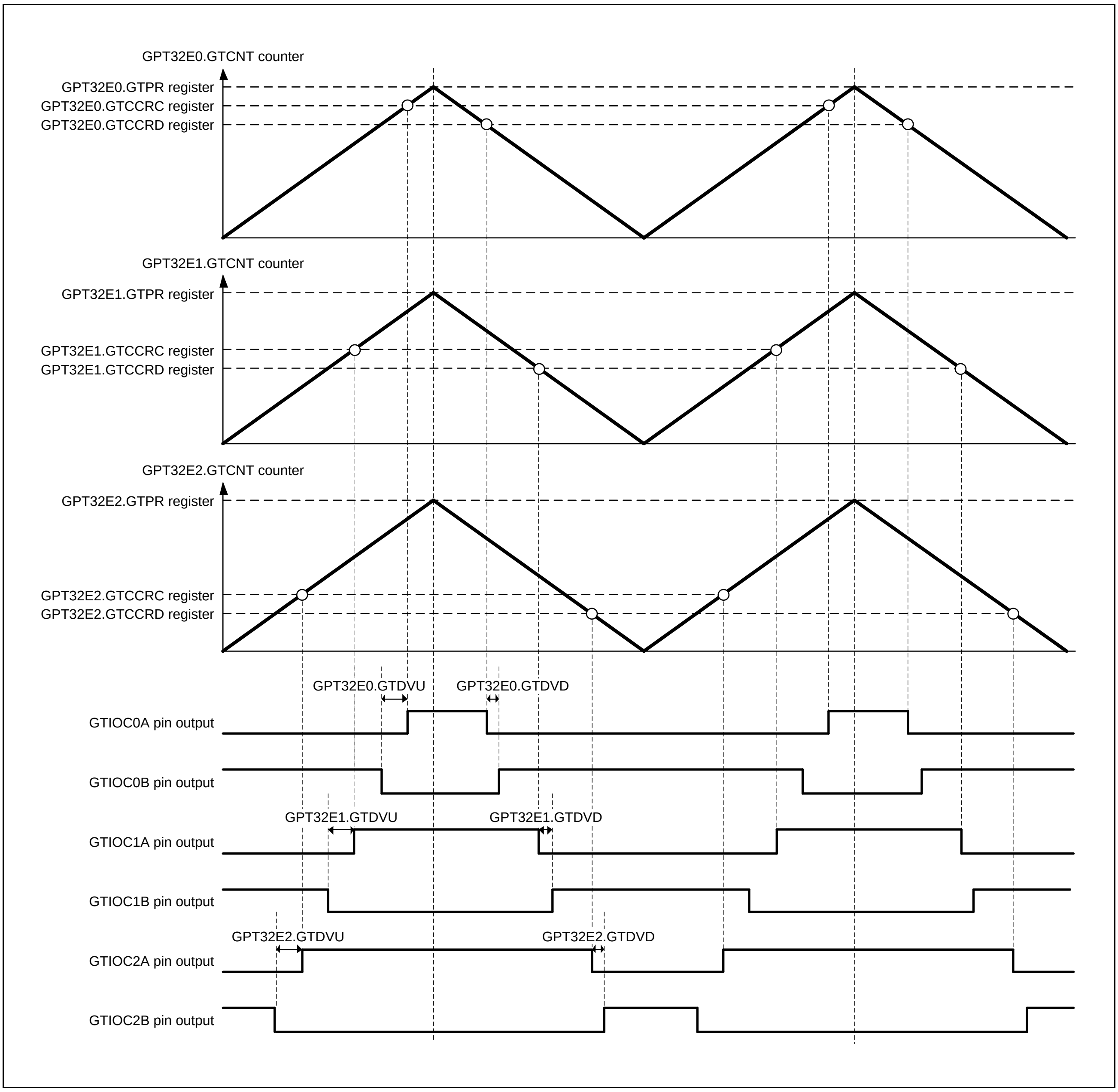


Figure 18.69 Example of 3-phase asymmetric triangle-wave complementary PWM output with automatic dead time setting

18.3.10 Phase Counting Function

The phase difference between the GTIOCA pin input and GTIOCB pin input is detected and the associated GTCNT counts up or counts down. The detectable phase difference is available in any combination with the relationship between the edge and the level of GTIOCA pin and GTIOCB pin input being set in the GTUPSR and GTDNSR registers. For details on count operation, see **Section 18.3.1.1, Counter Operation**.

Figure 18.70 to **Figure 18.79** show phase counting modes 1 to 5. **Table 18.8** to **Table 18.17** show conditions of up- counting or down-counting and lists settings for the GTUPSR and GTDNSR registers.

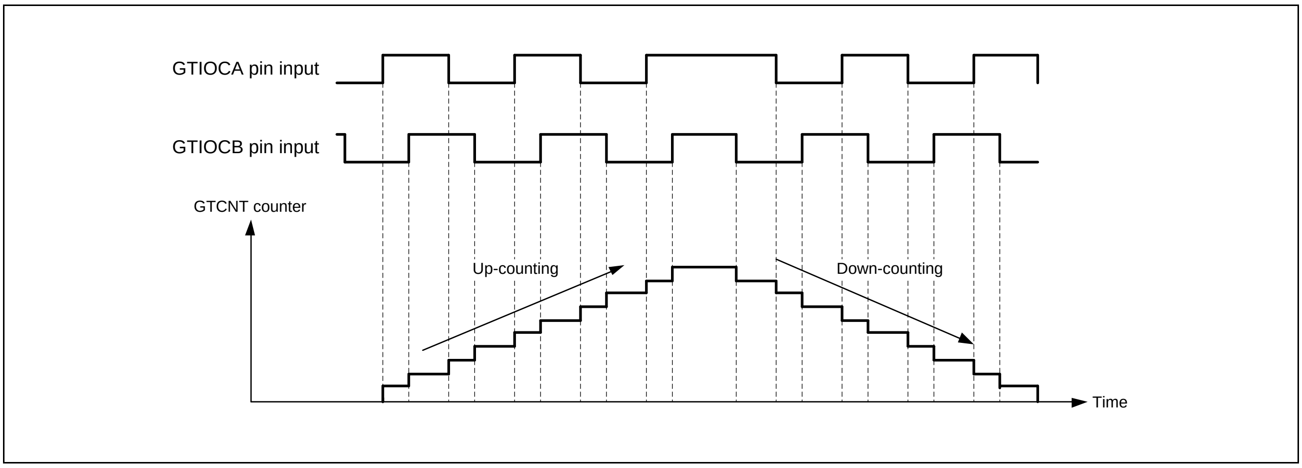


Figure 18.70 Example of phase counting mode 1

Table 18.8 Conditions of up-counting and down-counting in phase counting mode 1

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Up-counting	GTUPSR = H’00006900 GTDNSR = H’00009600
Low	↓		
↑	Low		
↓	High		
High	↓	Down-counting	
Low	↑		
↑	High		
↓	Low		

Remarks: ↑: Rising edge
↓: Falling edge

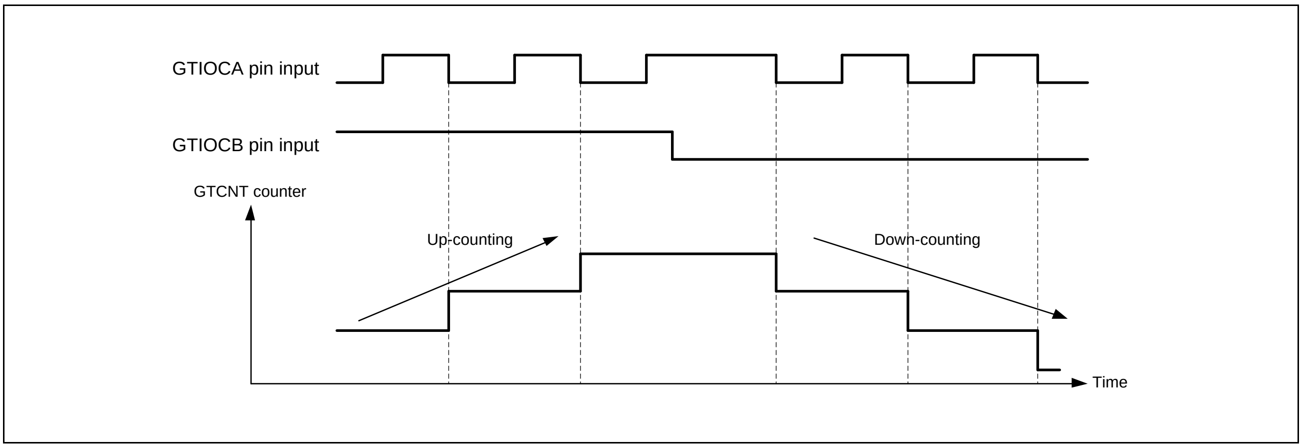


Figure 18.71 Example of phase counting mode 2 (A)

Table 18.9 Conditions of up-counting and down-counting in phase counting mode 2 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don’t care	GTUPSR = H’00000800 GTDNSR = H’00000400
Low	↓		
↑	Low		
↓	High	Up-counting	
High	↓	Don’t care	
Low	↑		
↑	High		
↓	Low	Down-counting	

Remarks: ↑ : Rising edge
↓ : Falling edge

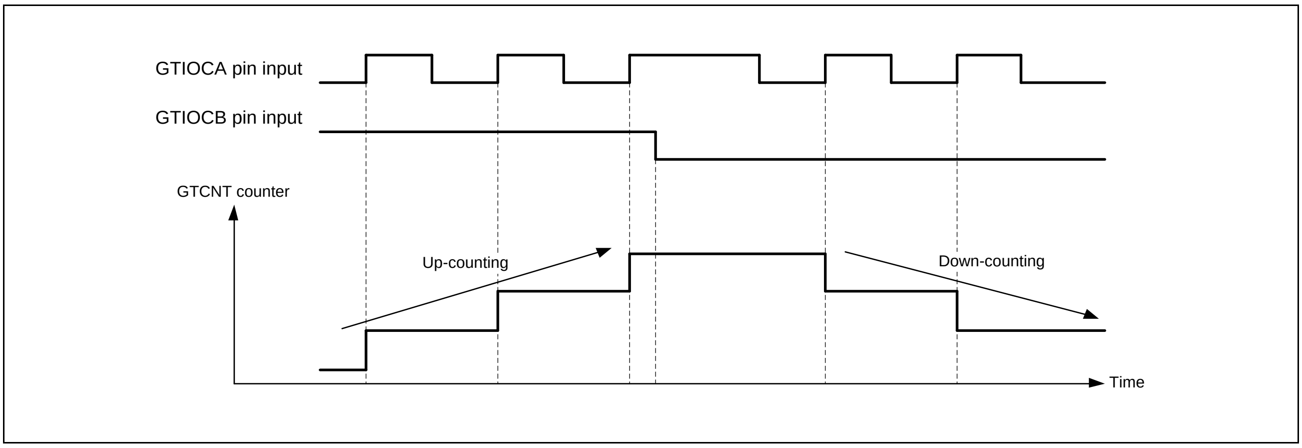


Figure 18.72 Example of phase counting mode 2 (B)

Table 18.10 Conditions of up-counting and down-counting in phase counting mode 2 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'00000200 GTDNSR = H'00000100
Low	↓		
↑	Low	Down-counting	
↓	High	Don't care	
High	↓		
Low	↑		
↑	High	Up-counting	
↓	Low	Don't care	

Remarks: ↑ : Rising edge
↓ : Falling edge

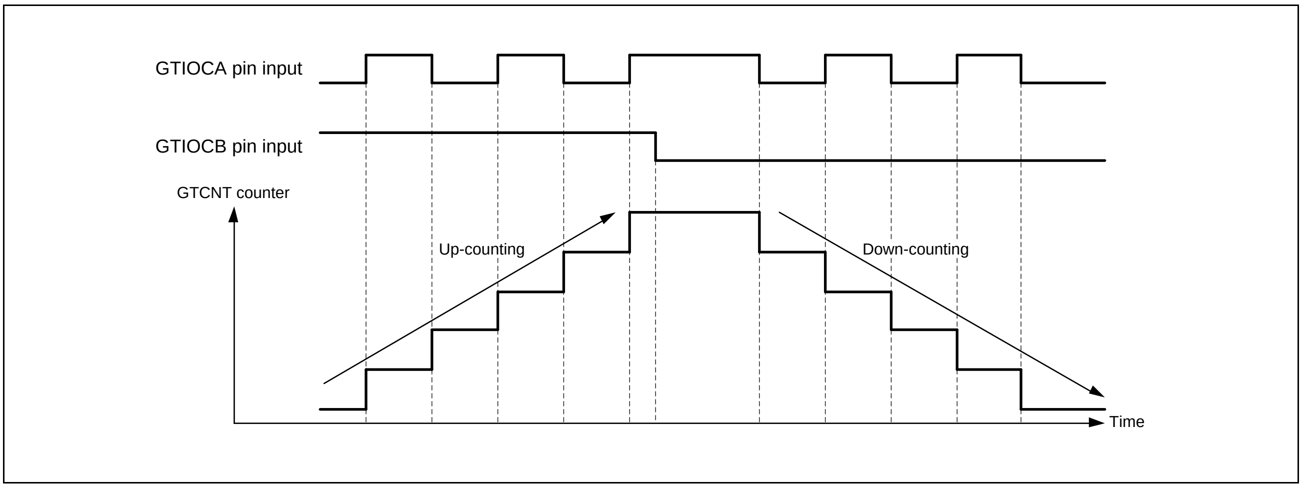


Figure 18.73 Example of phase counting mode 2 (C)

Table 18.11 Conditions of up-counting and down-counting in phase counting mode 2 (C)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don’t care	GTUPSR = H’00000A00 GTDNSR = H’00000500
Low	↓		
↑	Low	Down-counting	
↓	High	Up-counting	
High	↓	Don’t care	
Low	↑		
↑	High	Up-counting	
↓	Low	Down-counting	

Remarks: ↑ : Rising edge
↓ : Falling edge

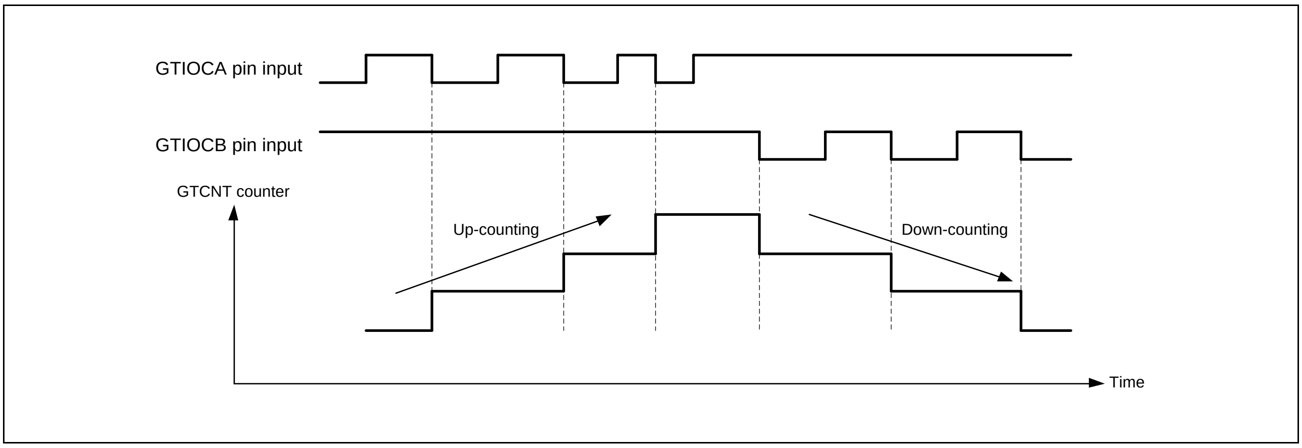


Figure 18.74 Example of phase counting mode 3 (A)

Table 18.12 Conditions of up-counting and down-counting in phase counting mode 3 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don’t care	GTUPSR = H’00000800 GTDNSR = H’00008000
Low	↓		
↑	Low		
↓	High	Up-counting	
High	↓	Down-counting	
Low	↑	Don’t care	
↑	High		
↓	Low		

Remarks: ↑ : Rising edge
↓ : Falling edge

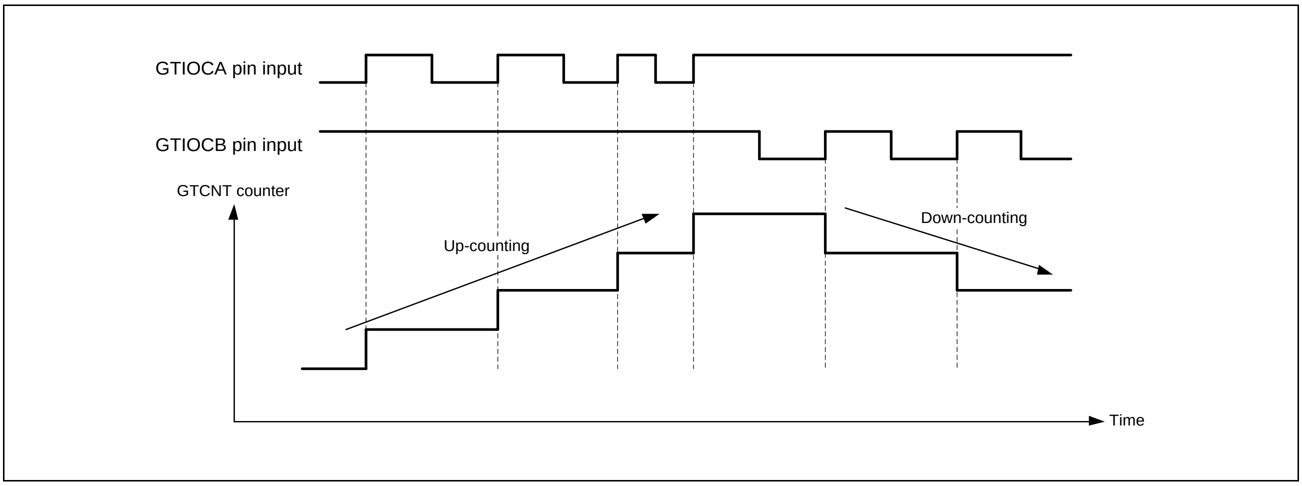


Figure 18.75 Example of phase counting mode 3 (B)

Table 18.13 Conditions of up-counting and down-counting in phase counting mode 3 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Down-counting	GTUPSR = H'00000200 GTDNSR = H'00002000
Low	↓	Don't care	
↑	Low		
↓	High		
High	↓		
Low	↑		
↑	High	Up-counting	
↓	Low	Don't care	

Remarks: ↑ : Rising edge
↓ : Falling edge

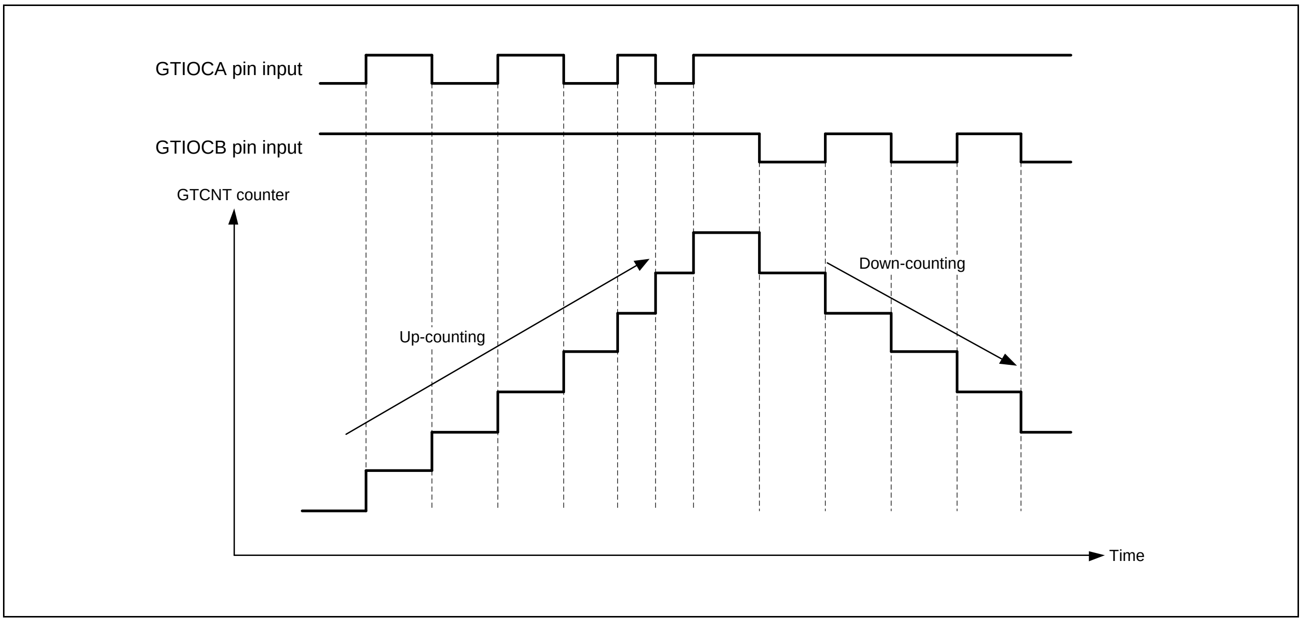


Figure 18.76 Example of phase counting mode 3 (C)

Table 18.14 Conditions of up-counting and down-counting in phase counting mode 3 (C)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Down-counting	GTUPSR = H’00000A00 GTDNSR = H’0000A000
Low	↓	Don’t care	
↑	Low		
↓	High	Up-counting	
High	↓	Down-counting	
Low	↑	Don’t care	
↑	High	Up-counting	
↓	Low	Don’t care	

Remarks: ↑ : Rising edge
↓ : Falling edge

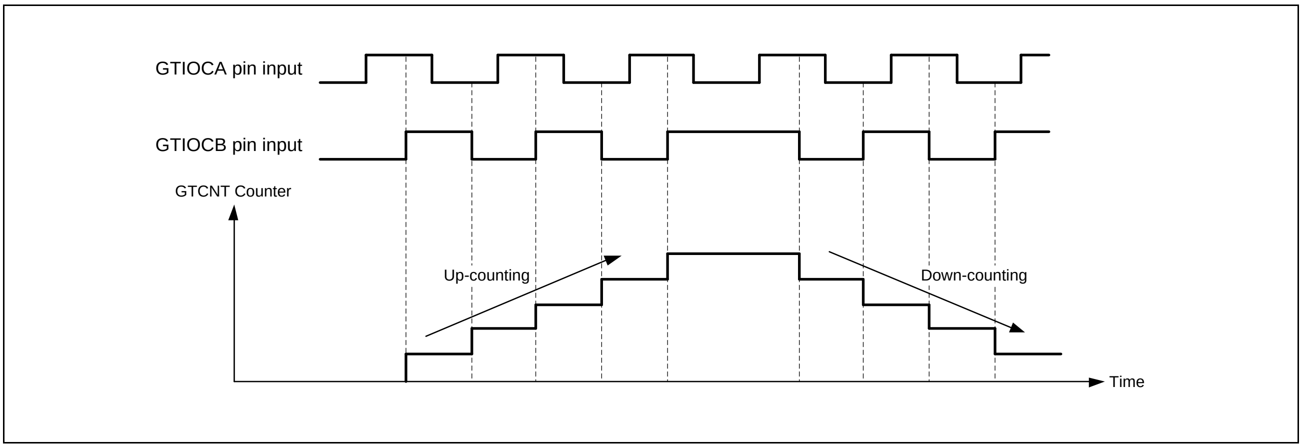


Figure 18.77 Example of phase counting mode 4

Table 18.15 Conditions of up-counting and down-counting in phase counting mode 4

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Up-counting	GTUPSR = H'00006000 GTDNSR = H'00009000
Low	↓		
↑	Low	Don't care	
↓	High		
High	↓	Down-counting	
Low	↑		
↑	High	Don't care	
↓	Low		

Remarks: ↑ : Rising edge
↓ : Falling edge

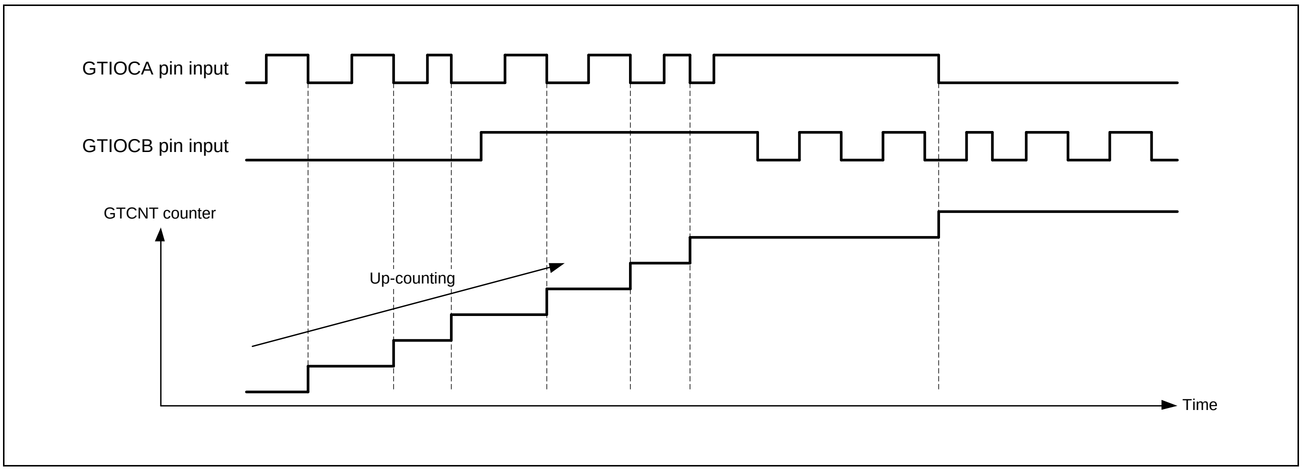


Figure 18.78 Example of phase counting mode 5 (A)

Table 18.16 Conditions of up-counting and down-counting in phase counting mode 5 (A)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don't care	GTUPSR = H'00000C00 GTDNSR = H'00000000
Low	↓		
↑	Low		
↓	High	Up-counting	
High	↓	Don't care	
Low	↑		
↑	High		
↓	Low	Up-counting	

Remarks: ↑ : Rising edge
↓ : Falling edge

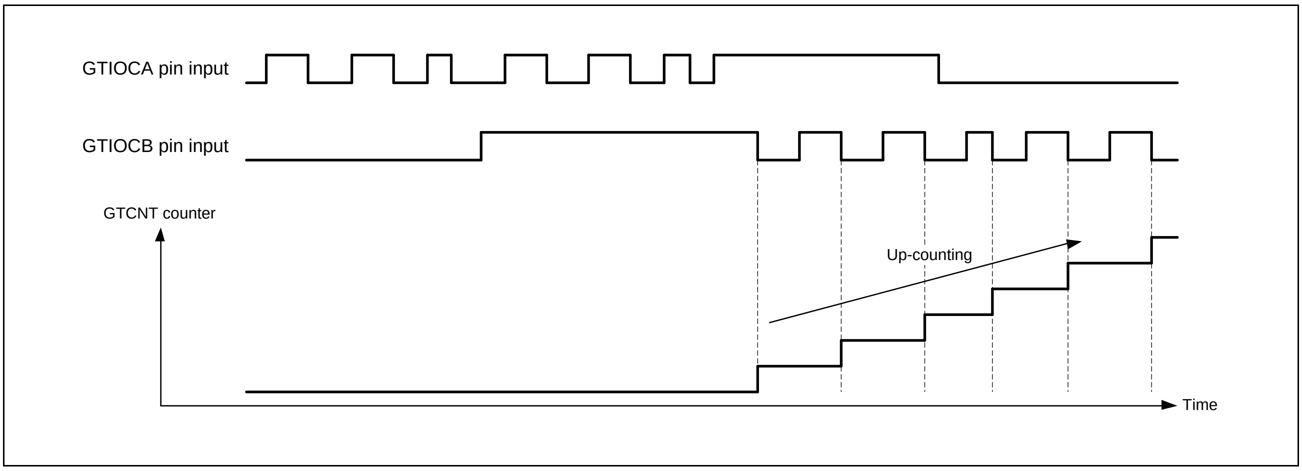


Figure 18.79 Example of phase counting mode 5 (B)

Table 18.17 Conditions of up-counting and down-counting in phase counting mode 5 (B)

GTIOCA pin input	GTIOCB pin input	Operation	Register setting
High	↑	Don’t care	GTUPSR = H’0000C000 GTDNSR = H’00000000
Low	↓	Up-counting	
↑	Low	Don’t care	
↓	High		
High	↓	Up-counting	
Low	↑	Don’t care	
↑	High		
↓	Low		

Remarks: ↑ : Rising edge
↓ : Falling edge

18.4 Interrupt Sources

18.4.1 Interrupt Sources and Priorities

The GPT provides the following interrupt sources:

- GTCCR input capture/compare match
- GTADTR compare match
- GTCNT counter overflow (GTPR compare match)/underflow.

Each interrupt source has its own status flag. When an interrupt source signal is generated, the associated status flag in GTST is set to 1. The associated status flag in GTST can be cleared by writing 0. If flag set and flag clear happen at the same time, flag clear takes priority over flag set. These flags are automatically updated by internal state. Relative channel priorities can be changed by the Interrupt Controller. However the priority within a channel is fixed. For details, see **Section 8, Interrupt Controller**. **Table 18.18** lists the GPT interrupt sources.

Table 18.18 Interrupt sources (1/3)

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
0	CCMPA0	GPT32E0.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB0	GPT32E0.GTCCRB input capture/compare match	TCFB	Possible
	CMPC0	GPT32E0.GTCCRC compare match	TCFC	Possible
	CMPD0	GPT32E0.GTCCRD compare match	TCFD	Possible
	CMPE0	GPT32E0.GTCCRE compare match	TCFE	Possible
	CMPF0	GPT32E0.GTCCRF compare match	TCFF	Possible
	ADTRGA0	GPT32E0.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB0	GPT32E0.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF0	GPT32E0.GTCNT overflow (GPT32E0.GTPR compare match)	TCFPO	Possible
	UNF0	GPT32E0.GTCNT underflow	TCFPU	Possible
1	CCMPA1	GPT32E1.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB1	GPT32E1.GTCCRB input capture/compare match	TCFB	Possible
	CMPC1	GPT32E1.GTCCRC compare match	TCFC	Possible
	CMPD1	GPT32E1.GTCCRD compare match	TCFD	Possible
	CMPE1	GPT32E1.GTCCRE compare match	TCFE	Possible
	CMPF1	GPT32E1.GTCCRF compare match	TCFF	Possible
	ADTRGA1	GPT32E1.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB1	GPT32E1.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF1	GPT32E1.GTCNT overflow (GPT32E1.GTPR compare match)	TCFPO	Possible
	UNF1	GPT32E1.GTCNT underflow	TCFPU	Possible

Table 18.18 Interrupt sources (2/3)

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
2	CCMPA2	GPT32E2.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB2	GPT32E2.GTCCRB input capture/compare match	TCFB	Possible
	CMPC2	GPT32E2.GTCCRC compare match	TCFC	Possible
	CMPD2	GPT32E2.GTCCRD compare match	TCFD	Possible
	CMPE2	GPT32E2.GTCCRE compare match	TCFE	Possible
	CMPF2	GPT32E2.GTCCRF compare match	TCFF	Possible
	ADTRGA2	GPT32E2.GTCCRE compare match	ADTRAUF ADTRADF	Possible
	ADTRGB2	GPT32E2.GTCCRF compare match	ADTRBUF ADTRBDF	Possible
	OVF2	GPT32E2.GTCNT overflow (GPT32E2.GTPR compare match)	TCFPO	Possible
	UNF2	GPT32E2.GTCNT underflow	TCFPU	Possible
3	CCMPA3	GPT32E3.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB3	GPT32E3.GTCCRB input capture/compare match	TCFB	Possible
	CMPC3	GPT32E3.GTCCRC compare match	TCFC	Possible
	CMPD3	GPT32E3.GTCCRD compare match	TCFD	Possible
	CMPE3	GPT32E3.GTCCRE compare match	TCFE	Possible
	CMPF3	GPT32E3.GTCCRF compare match	TCFF	Possible
	ADTRGA3	GPT32E3.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB3	GPT32E3.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF3	GPT32E3.GTCNT overflow (GPT32E3.GTPR compare match)	TCFPO	Possible
	UNF3	GPT32E3.GTCNT underflow	TCFPU	Possible
4	CCMPA4	GPT32E4.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB4	GPT32E4.GTCCRB input capture/compare match	TCFB	Possible
	CMPC4	GPT32E4.GTCCRC compare match	TCFC	Possible
	CMPD4	GPT32E4.GTCCRD compare match	TCFD	Possible
	CMPE4	GPT32E4.GTCCRE compare match	TCFE	Possible
	CMPF4	GPT32E4.GTCCRF compare match	TCFF	Possible
	ADTRGA4	GPT32E4.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB4	GPT32E4.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF4	GPT32E4.GTCNT overflow (GPT32E4.GTPR compare match)	TCFPO	Possible
	UNF4	GPT32E4.GTCNT underflow	TCFPU	Possible

Table 18.18 Interrupt sources (3/3)

Channel	Name	Interrupt source	Interrupt flag	DMAC activation
5	CCMPA5	GPT32E5.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB5	GPT32E5.GTCCRB input capture/compare match	TCFB	Possible
	CMPC5	GPT32E5.GTCCRC compare match	TCFC	Possible
	CMPD5	GPT32E5.GTCCRD compare match	TCFD	Possible
	CMPE5	GPT32E5.GTCCRE compare match	TCFE	Possible
	CMPF5	GPT32E5.GTCCRF compare match	TCFF	Possible
	ADTRGA5	GPT32E5.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB5	GPT32E5.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF5	GPT32E5.GTCNT overflow (GPT32E5.GTPR compare match)	TCFPO	Possible
	UNF5	GPT32E5.GTCNT underflow	TCFPU	Possible
6	CCMPA6	GPT32E6.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB6	GPT32E6.GTCCRB input capture/compare match	TCFB	Possible
	CMPC6	GPT32E6.GTCCRC compare match	TCFC	Possible
	CMPD6	GPT32E6.GTCCRD compare match	TCFD	Possible
	CMPE6	GPT32E6.GTCCRE compare match	TCFE	Possible
	CMPF6	GPT32E6.GTCCRF compare match	TCFF	Possible
	ADTRGA6	GPT32E6.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB6	GPT32E6.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF6	GPT32E6.GTCNT overflow (GPT32E6.GTPR compare match)	TCFPO	Possible
	UNF6	GPT32E6.GTCNT underflow	TCFPU	Possible
7	CCMPA7	GPT32E7.GTCCRA input capture/compare match	TCFA	Possible
	CCMPB7	GPT32E7.GTCCRB input capture/compare match	TCFB	Possible
	CMPC7	GPT32E7.GTCCRC compare match	TCFC	Possible
	CMPD7	GPT32E7.GTCCRD compare match	TCFD	Possible
	CMPE7	GPT32E7.GTCCRE compare match	TCFE	Possible
	CMPF7	GPT32E7.GTCCRF compare match	TCFF	Possible
	ADTRGA7	GPT32E7.GTADTRA compare match	ADTRAUF ADTRADF	Possible
	ADTRGB7	GPT32E7.GTADTRB compare match	ADTRBUF ADTRBDF	Possible
	OVF7	GPT32E7.GTCNT overflow (GPT32E7.GTPR compare match)	TCFPO	Possible
	UNF7	GPT32E7.GTCNT underflow	TCFPU	Possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

(1) ADTRGAn interrupt (n = 0 to 7)

When the GTCNT counter value matches with the GTADTRA register, an interrupt request is generated under the following condition:

- In up-counting, the interrupt enable bit (ADTRAUEN) in the GTINTAD register is 1
- In down-counting, the interrupt enable bit (ADTRADEN) in the GTINTAD register is 1.

In event count operation performing, this interrupt request is not generated.

(2) ADTRGBn interrupt (n = 0 to 7)

When the GTCNT counter value matches with the GTADTRB register, an interrupt request is generated under the following condition:

- In up-counting, the interrupt enable bit (ADTRBUEN) in the GTINTAD register is 1
- In down-counting, the interrupt enable bit (ADTRBDEN) in the GTINTAD register is 1.

In event count operation performing, this interrupt request is not generated.

(3) CCMPAn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRA register functions as a compare match register, the GTCNT counter value matches with the GTCCRA register.
- When the GTCCRA register functions as an input capture register, the GTCNT counter value is transferred to the GTCCRA register by an input capture signal.

(4) CCMPBn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRB register functions as a compare match register, the GTCNT counter value matches with the GTCCRB register.
- When the GTCCRB register functions as an input capture register, the GTCNT counter value is transferred to the GTCCRB register by an input capture signal.

(5) CMPCn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRC register functions as a compare match register, the GTCNT counter value matches with the GTCCRC register.

A compare match is not performed and an interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (buffer operation with the GTCCRC register).

(6) CMPDn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRD register functions as a compare match register, the GTCNT counter value matches with the GTCCRD register.

A compare match is not performed and therefore interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (buffer operation with the GTCCRD register).

(7) CMPEn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition.

- When the GTCCRE register functions as a compare match register, the GTCNT counter value matches with the GTCCRE register.

A compare match is not performed and an interrupt is not requested under the following conditions.

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (buffer operation with the GTCCRE register)

(8) CMPFn interrupt (n = 0 to 7)

An interrupt request is generated under the following condition:

- When the GTCCRF register functions as a compare match register, the GTCNT counter value matches with the GTCCRF register.

A compare match is not performed and therefore interrupt is not requested under the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (buffer operation with the GTCCRF register).

(9) OVFn interrupt (n = 0 to 7)

An interrupt request is generated under the following conditions:

- In saw-wave mode, interrupt requests are enabled at overflows (when the GTCNT counter value changes from GTPR to 0 during up-counting)
- In triangle-wave mode, interrupt requests are enabled at crests (GTCNT changes from GTPR to GTPR-1)
- In counting by hardware sources, overflow (GTCNT changes from GTPR to 0 in up count) has occurred.

(10) UNFn interrupt (n = 0 to 7)

An interrupt request is generated under the following conditions:

- In saw-wave mode, interrupt requests are enabled at underflows (when the GTCNT counter value changes from 0 to GTPR during down-counting).
- In triangle-wave mode, interrupt requests are enabled at troughs (GTCNT changes from 0 to 1).
- In counting by hardware sources, underflow (GTCNT changes from 0 to GTPR in down count) has occurred.

Table 18.19 Interrupt signals, interrupt permission bits, and interrupt status flags

Interrupt signal	Interrupt permission bit	Interrupt status flag
UNFn	GTINTAD[7:6] (GTINTPR[1:0])	GTST[7] (TCFPU)
OVFn		GTST[6] (TCFPO)
ADTRGBn	GTINTAD[19] (ADTRBDEN) GTINTAD[18] (ADTRBUEN)	GTST[19] (ADTRBDF) GTST[18] (ADTRBUF)
ADTRGAn	GTINTAD[17] (ADTRADEN) GTINTAD[16] (ADTRAUEN)	GTST[17] (ADTRADF) GTST[16] (ADTRAUF)
CMPFn	GTINTAD[5] (GTINTF)	GTST[5] (TCFF)
CMPEn	GTINTAD[4] (GTINTE)	GTST[4] (TCFE)
CMPDn	GTINTAD[3] (GTINTD)	GTST[3] (TCFD)
CMPCn	GTINTAD[2] (GTINTC)	GTST[2] (TCFC)
CCMPBn	GTINTAD[1] (GTINTB)	GTST[1] (TCFB)
CCMPAn	GTINTAD[0] (GTINTA)	GTST[0] (TCFA)

18.4.2 DMAC Activation

The DMAC can be activated by the interrupt in each channel. For details, see **Section 14, Direct Memory Access Controller**.

18.4.3 Interrupt and A/D Conversion Request Skipping Function

By setting the GTITC register, the GTCNT counter overflow (GTPR compare match) interrupt (OVFn) and underflow interrupt (UNFn) can be skipped. Other interrupts and A/D converter start request signals can be skipped in coordination with the OVFn/UNFn skipping function.

The interrupt request skipping function only depends on the setting of GTITC register and is independent of the setting of interrupt permission bits in the GTINTAD register.

When both troughs and crests are counted and skipped in triangle-wave mode, if the number of times of skipping is odd, OVFn/UNFn interrupt requests cannot be generated at troughs only or at crests only depending on the skipping counter start timing. To count both troughs and crests and generate the OVFn/UNFn interrupts at troughs only or crests only in triangle-wave mode, you must set an even number of skips.

Similarly, in saw-wave mode, when both overflows and underflows are counted and skipped with the count direction changed, OVFn interrupt requests cannot be generated on either overflows or underflows only. To count both overflows and underflows with the count direction changed and generate the OVFn/UNFn interrupts on either overflows or underflows only in saw wave mode, first check the skipping state carefully.

Before changing the skipping count, you must release the skipping count setting (GTITC.IVTC[1:0] bits = 00b).

Figure 18.80 to **Figure 18.85** show examples of skipping function operation.

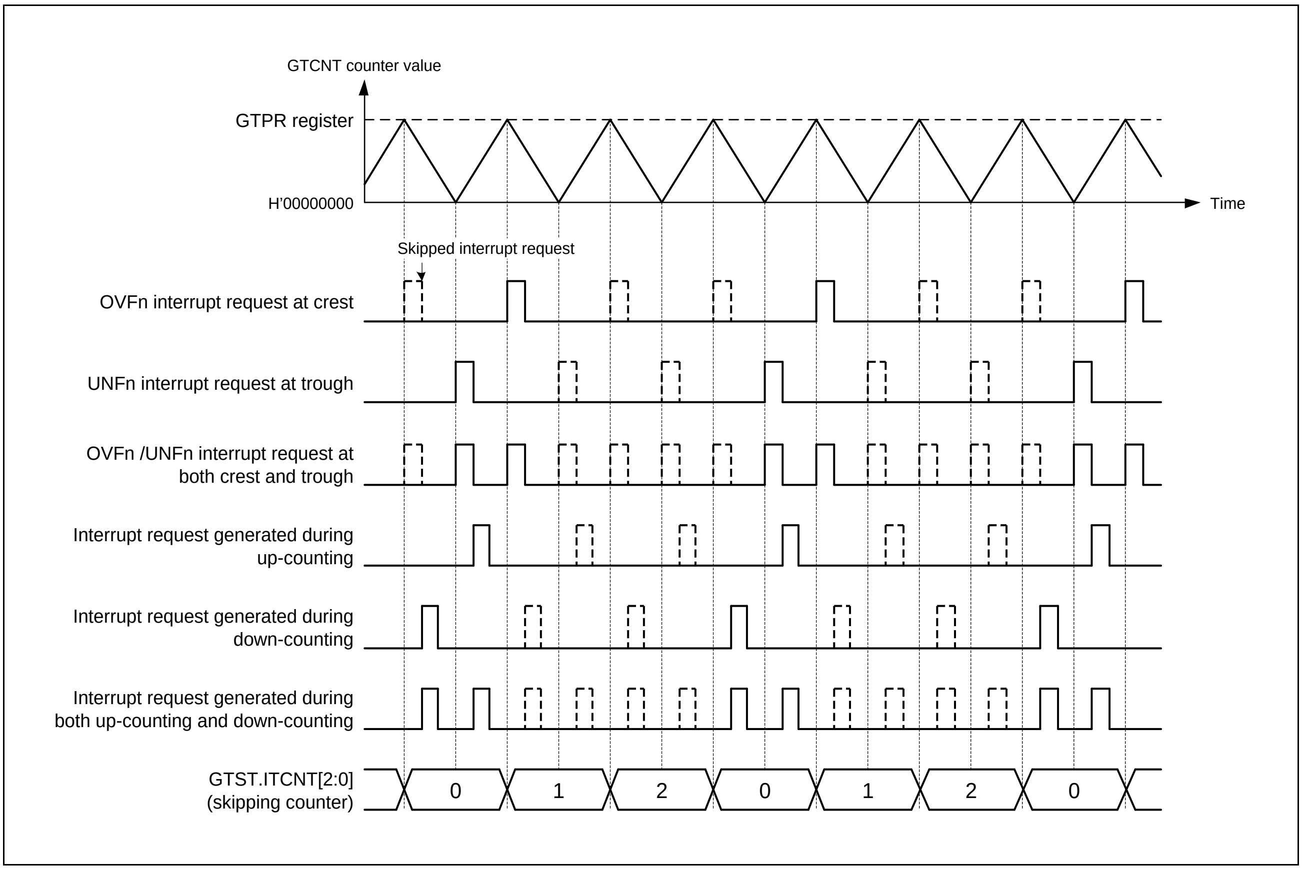


Figure 18.80 Example of interrupt skipping function operation with triangle waves, counting and skipping crests, and skipping count = 2

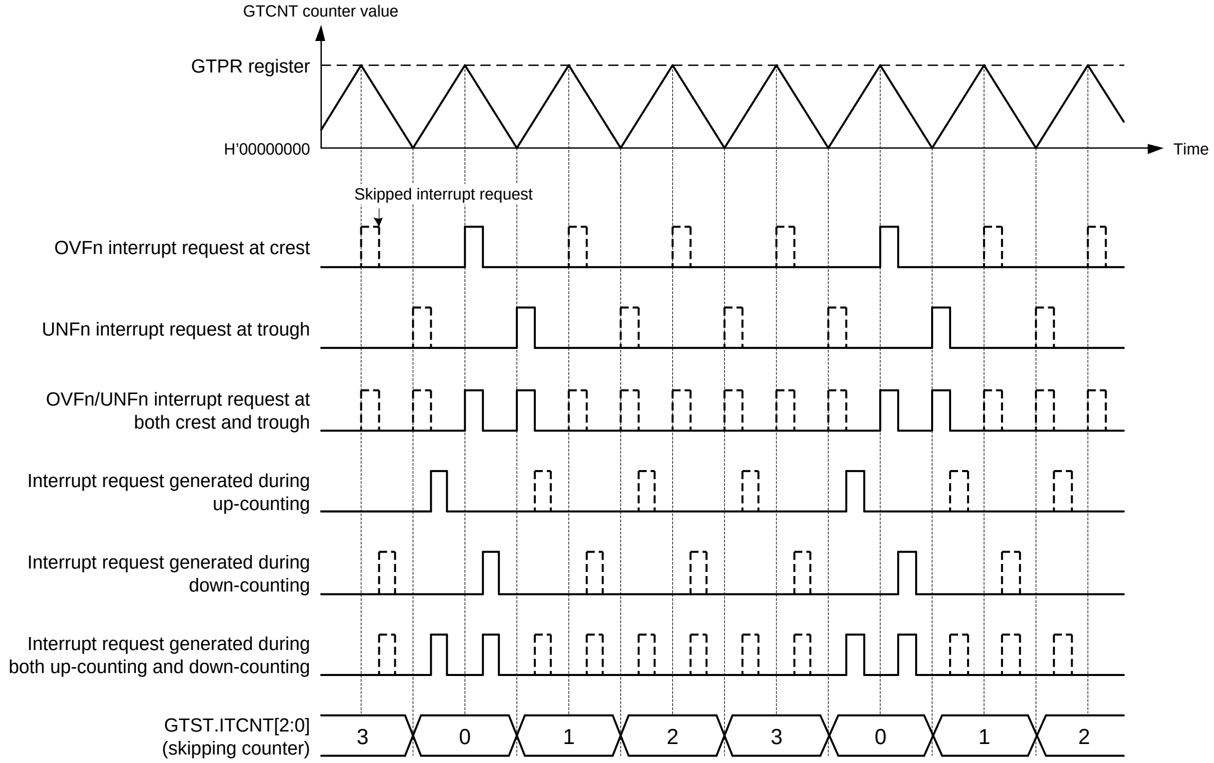


Figure 18.81 Example of interrupt skipping function operation with triangle waves, counting and skipping troughs, and skipping count = 3

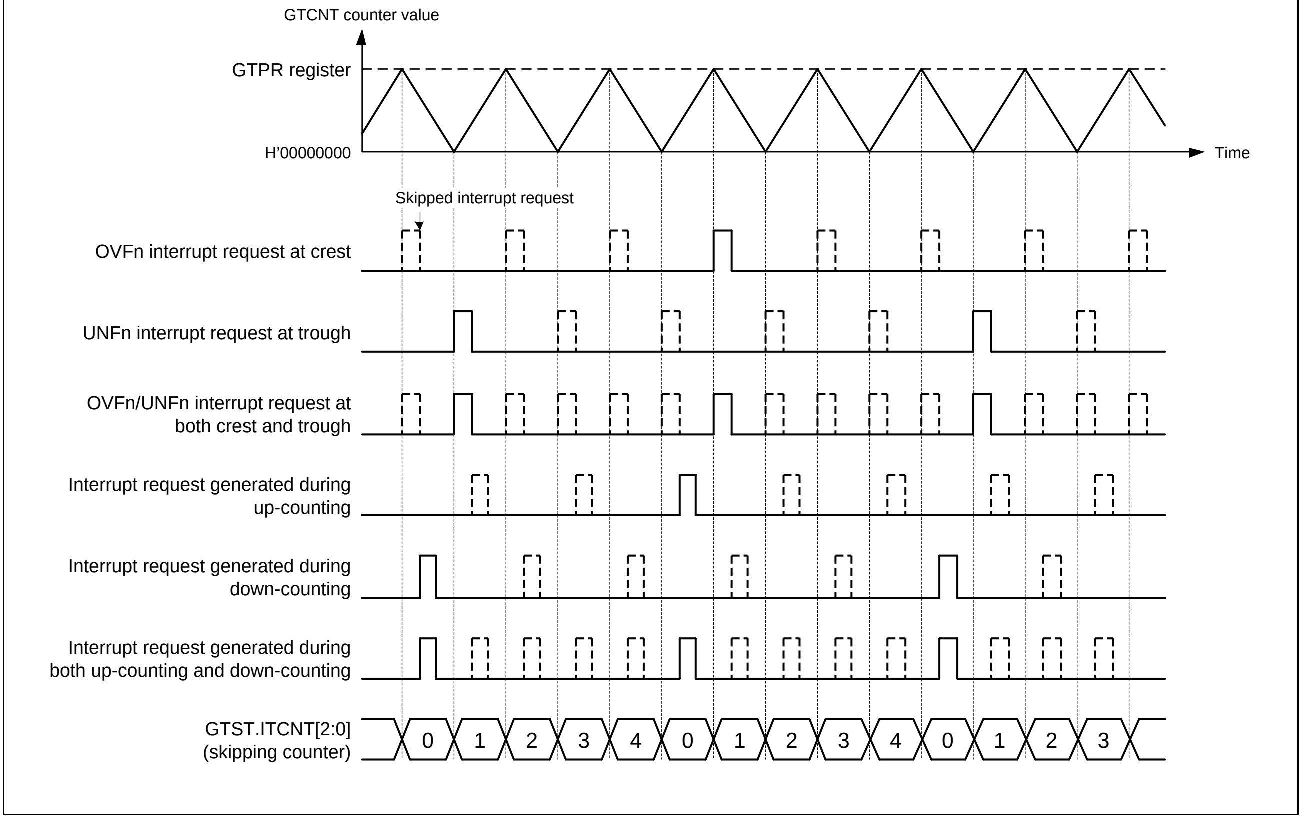


Figure 18.82 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, and skipping count = 4

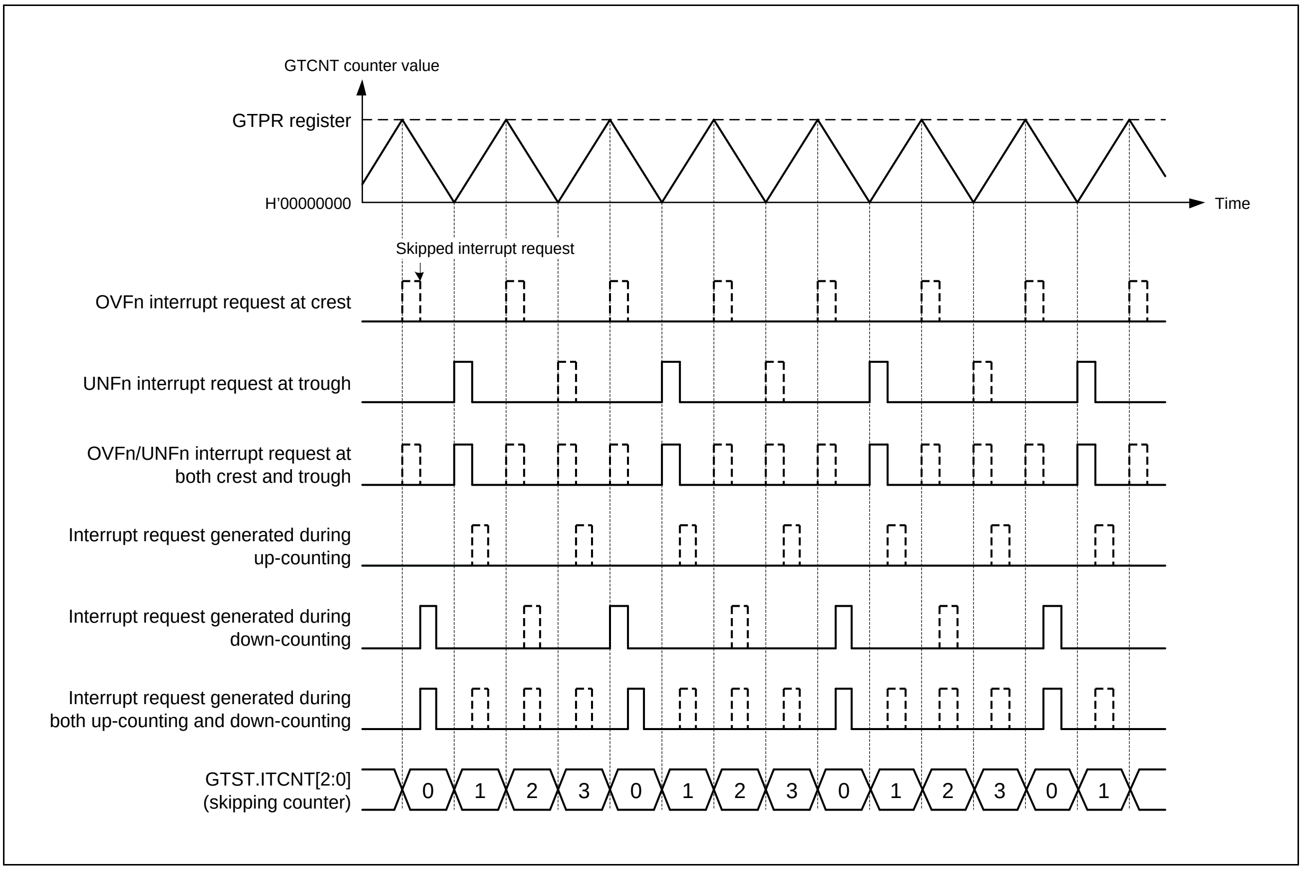


Figure 18.83 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, skipping count = 3, and skipping started at up-counting

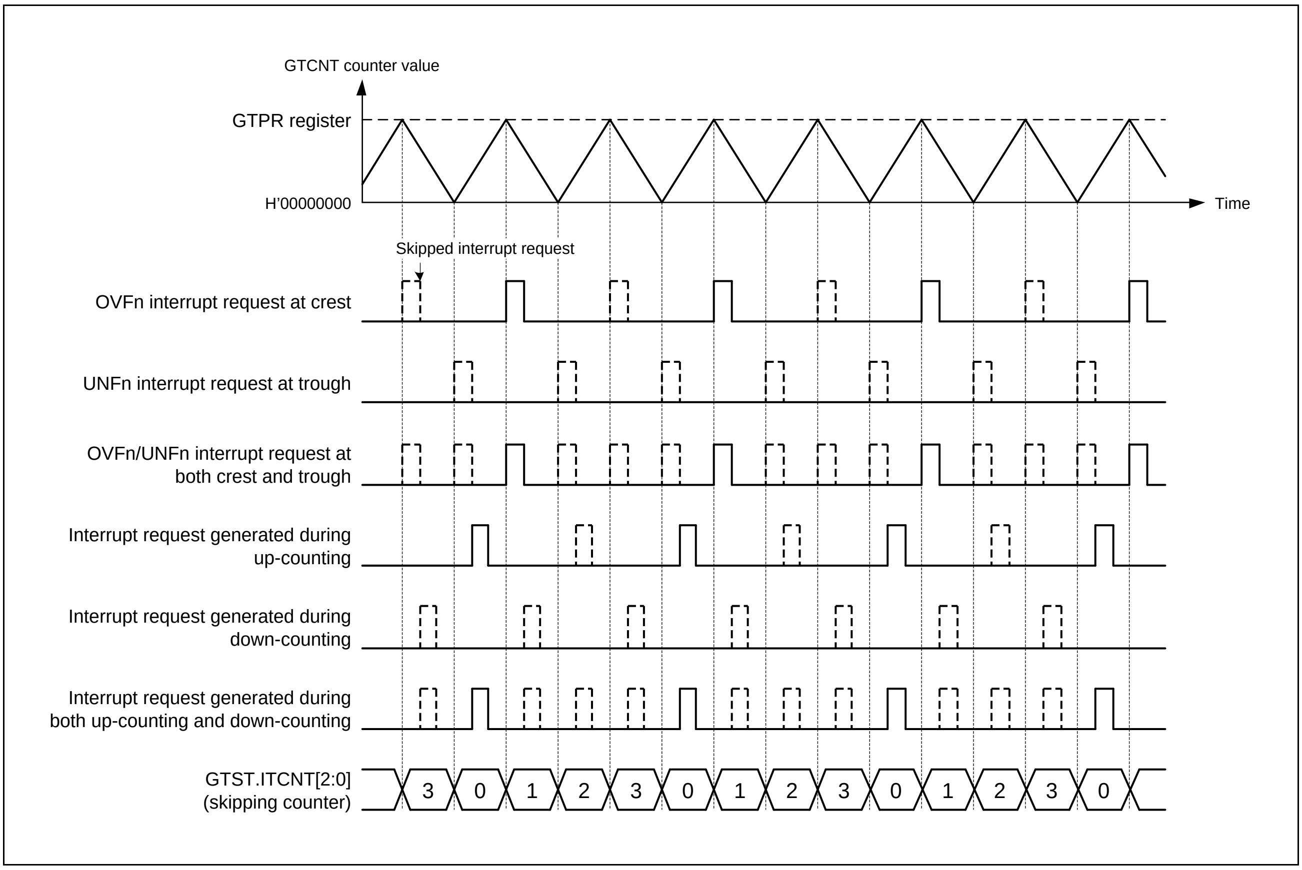


Figure 18.84 Example of interrupt skipping function operation with triangle waves, counting and skipping both troughs and crests, skipping count = 3, and skipping started at down-counting

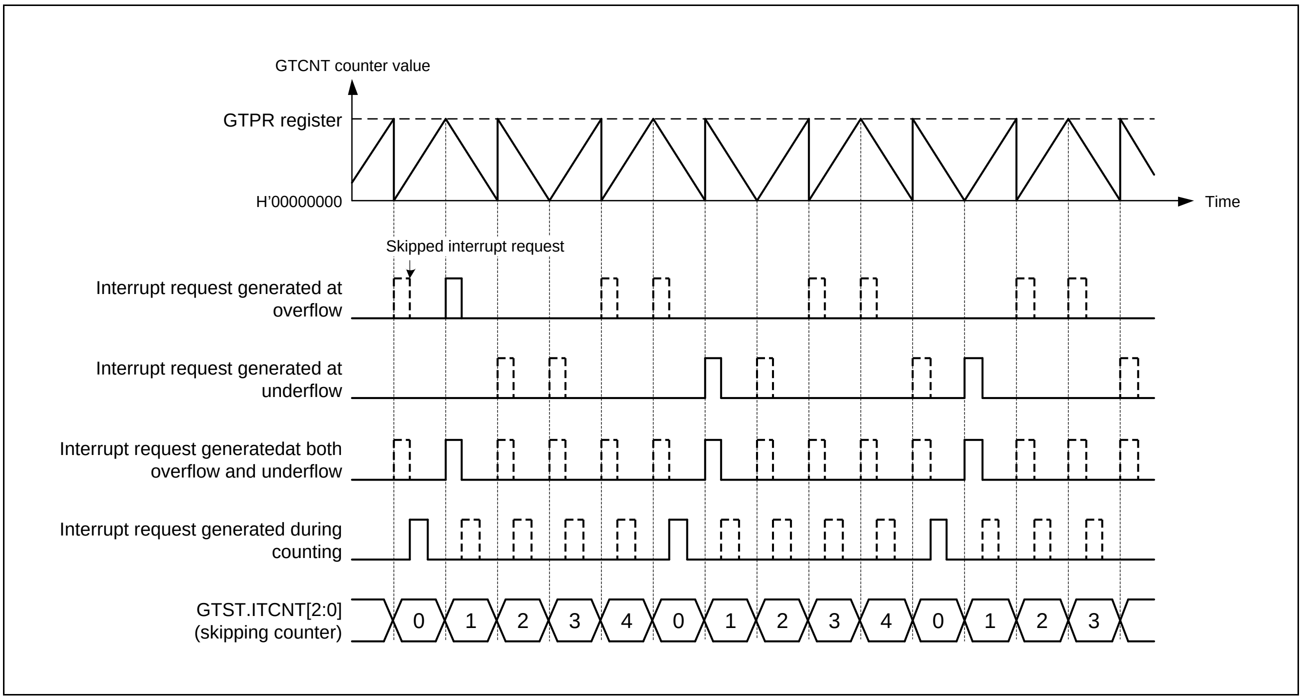


Figure 18.85 Example of interrupt skipping function operation with saw waves, operation with count direction changed, counting and skipping both overflows and underflows, and skipping count = 4

18.5 A/D Converter Start Request

An A/D converter start request can be issued at a compare match between the GTCNT counter and GTADTRA or GTADTRB, and up-counting only, down-counting only, or both up-counting and down-counting can be specified. In event count operation performing, A/D converter start requests interrupt cannot be generated.

GTADTRA and GTADTRB each has two buffer registers. Buffer operation with GTADTRA combined with GTADTBRA and GTADTDBRA, and buffer operation with GTADTRB combined with GTADTBRB and GTADTDBRB can be performed.

Figure 18.86 shows an example of A/D converter start request operation, and **Figure 18.87** shows an example setting for A/D converter start request operation.

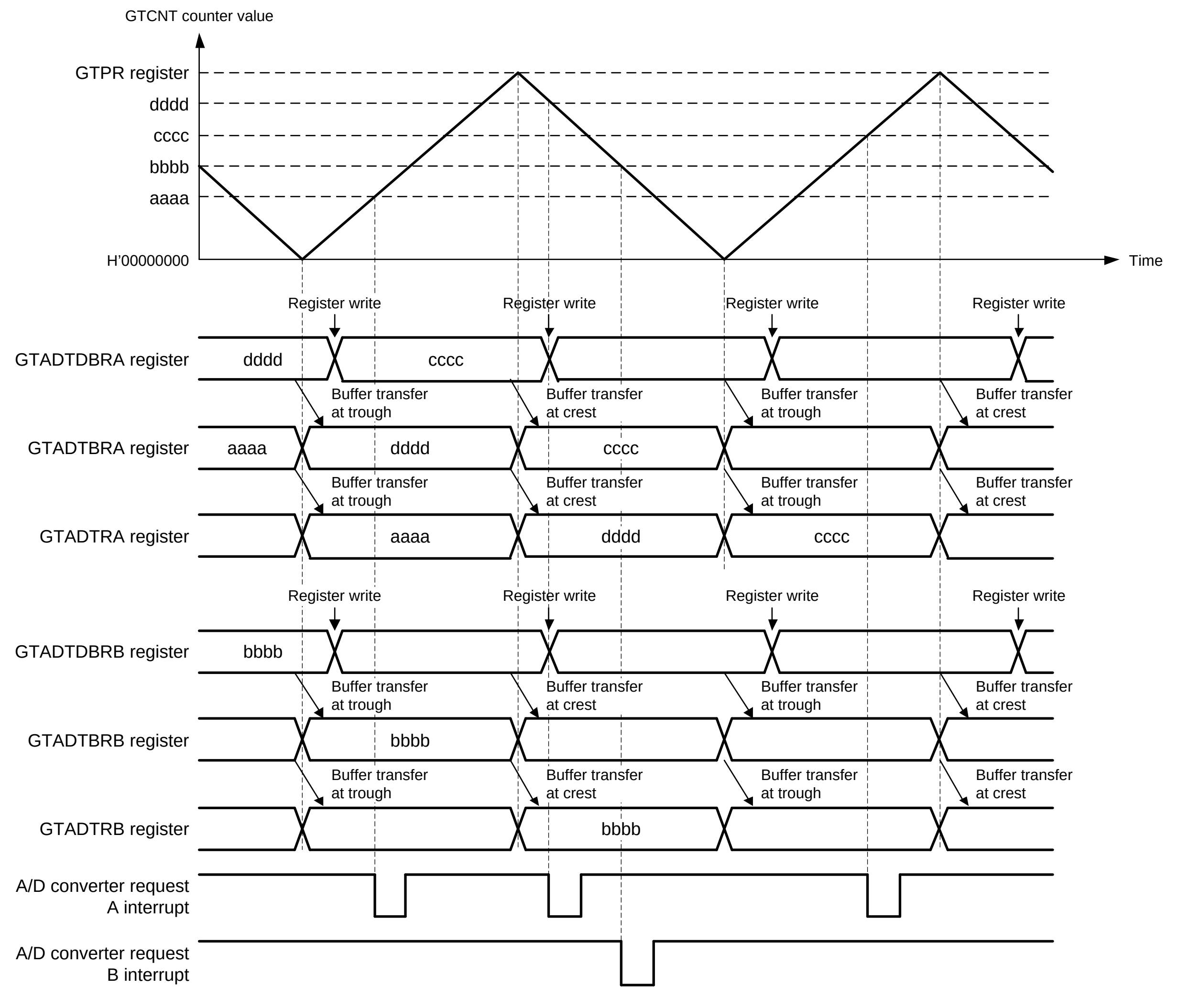


Figure 18.86 Example of A/D converter start request timing operation with triangle waves, double buffer operation, buffer transfer at both troughs and crests, A/D converter start request interrupt by GTADTRA at both up-counting and down-counting, and A/D converter start request interrupt by GTADTRB at down-counting

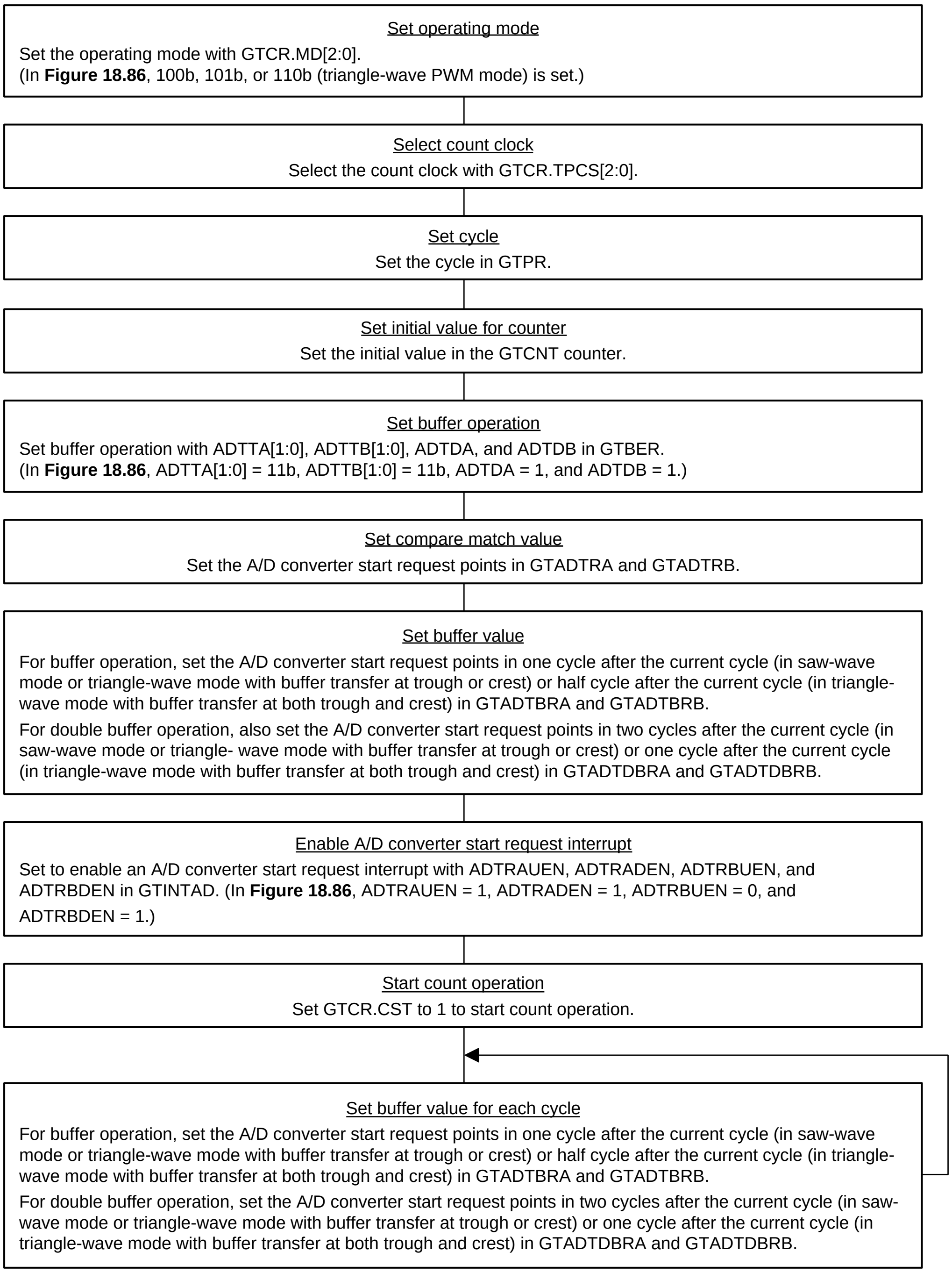


Figure 18.87 Example setting for A/D converter start request timing operation

18.6 Noise Filter Function

Each pin for use in input capture to the GPT is equipped with a noise filter. The noise filter samples input signals at the sampling clock and removes the pulses whose length is less than three sampling cycles.

The noise filter functionality includes enabling and disabling the noise filter for each pin and setting of the sampling clock for each channel.

Figure 18.88 shows the timing of noise filtering.

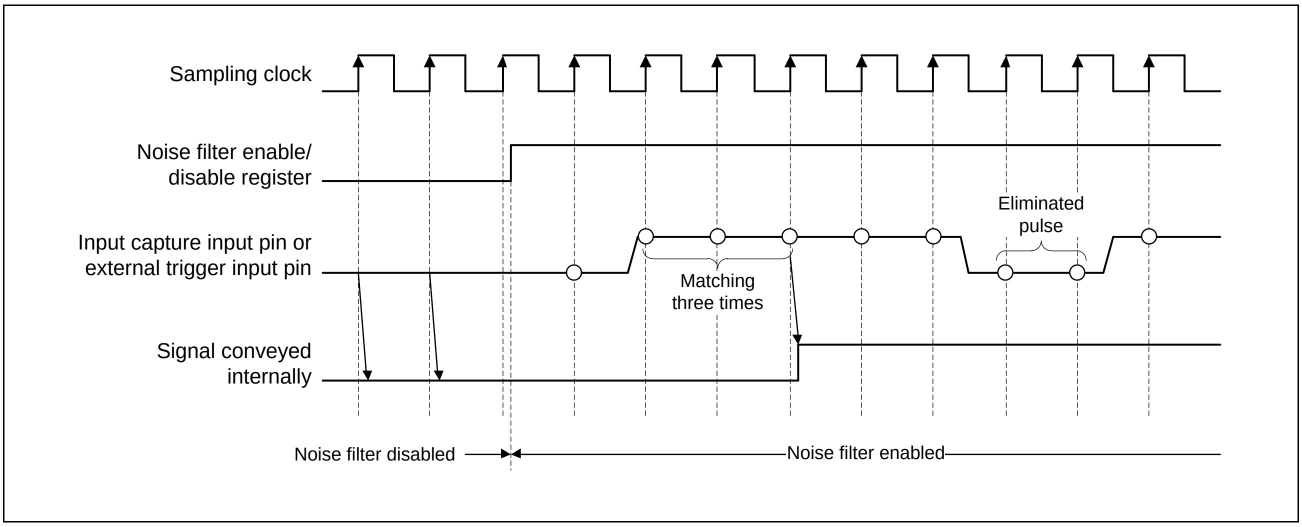


Figure 18.88 Timing of noise filtering

If noise filtering is enabled, the input capture operation or external trigger operation performs on the edges of the noise filtered signal after a delay of a sampling interval × 3 + P0ϕ. This is caused by the noise filtering for the input capture input or external trigger operation.

18.7 Protection Function

18.7.1 Write-Protection for Registers

To prevent registers from being accidentally modified, registers can be write-protected in channel units by setting GTWP.WP. Write-protection can be set for the following registers:

> GTSSR, GTPSR, GTCSR, GTUPSR, GTDNSR, GTICASR, GTICBSR, GTCR, GTUDDTYC, GTIOR, GTINTAD,GTST, GTBER, GTITC, GTCNT, GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, GTCCRF, GTPR, GTPBR, GTPDBR, GTADTRA, GTADTBRA, GTADTDBRA, GTADTRB, GTADTBRB, GTADTDBRB, GTDTCR, GTDVU, GTDVD, GTDBU, GTDBD, GTSOS, GTSOTR.

18.7.2 Disabling of Buffer Operation

If the timing of buffer register write is delayed in relative to the timing for the buffer transfer, buffer operation can be suspended with the GTBER.BD setting. Buffer transfer can be temporarily disabled even when a buffer transfer condition is generated during a buffer register write. This can be done by setting the associated GTBER.BD bit to 1 (buffer operation disabled) before a buffer register write and clearing the bit to 0 (buffer operation enabled) after completion of writing to all buffer registers. **Figure 18.89** shows an example of operation for disabling buffer operation.

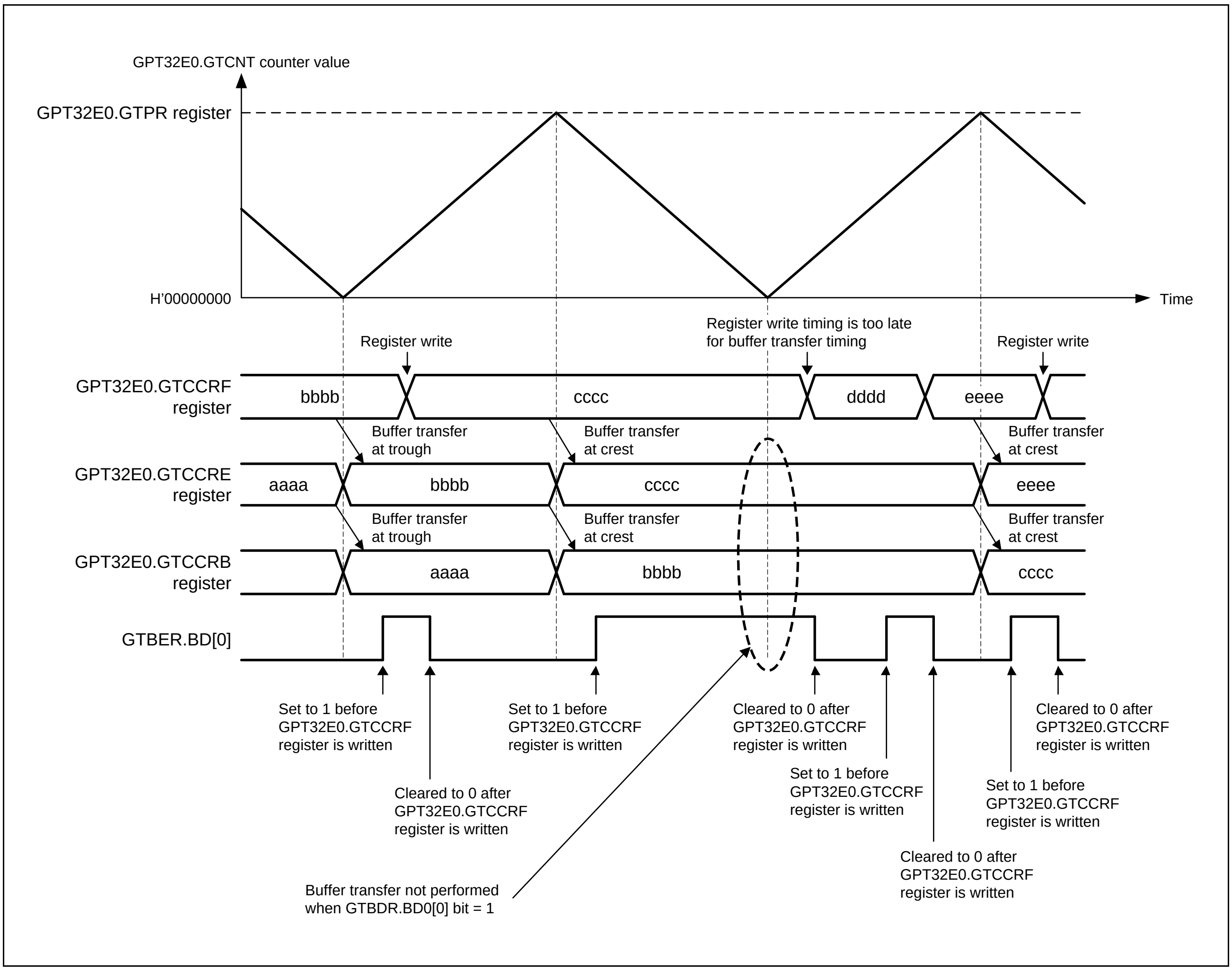


Figure 18.89 Example of operation for disabling buffer operation with triangle waves, double buffer operation, and buffer transfer at both troughs and crests

18.7.3 GTIOC Pin Output Negate Control

For protection from system failure, the output disable control that changes the GTIOC pin output value forcibly is provided for GTIOC pin output by the request of output disable from POEG.

When dead time error occurs or the GTIOCA pin output value is the same as the GTIOCB pin output value, output protection is required. GPT detects this condition and generates output disable requests to POEG based on the settings in the output disable request permission bits, such as GTINTAD.GRPDTE, GTINTAD.GRPABH, GTINTAD.GRPABL. After the POEG receives output disable requests from each channel and calculates external input using an OR operation, the POEG generates output disable requests to GPT.

One output disable signal (representing the shared output disable request signal of the GTIOCA pin and the GTIOCB pin) out of four* output disable requests generated by the POEG is selected by setting GTINTAD.GRP[1:0]. The status of the selected output disable request is monitored by reading the GTST.ODF bit. The output level during output disable is based on the GTIOR.OADF[1:0] setting for the GTIOCA pin and the GTIOR.OBDF[1:0] setting for the GTIOCB pin. The change to the output disable state is performed asynchronously by generating the output disable request from the POEG. The release of the output disable state is performed at end of cycle by terminating the output disable request. The timing of release of the output disable state is a minimum of 3 P0ϕ cycles after terminating the output disable request. To perform output disable control reliably, allow at least 4 P0ϕ cycles after generating the output disable request (by clearing the output disable request flag in POEG) until the output disable request is terminated.

When event count is performed or when the output disable state is to be released immediately without waiting for an end of cycle, GTIOR.OADF[1:0] must be set to 00b (for GTIOCA pin) or GTIOR.OBDF[1:0] must be set to 00b (for GTIOCB pin).

Figure 18.90 shows an example of the GTIOC pin output disable control operation.

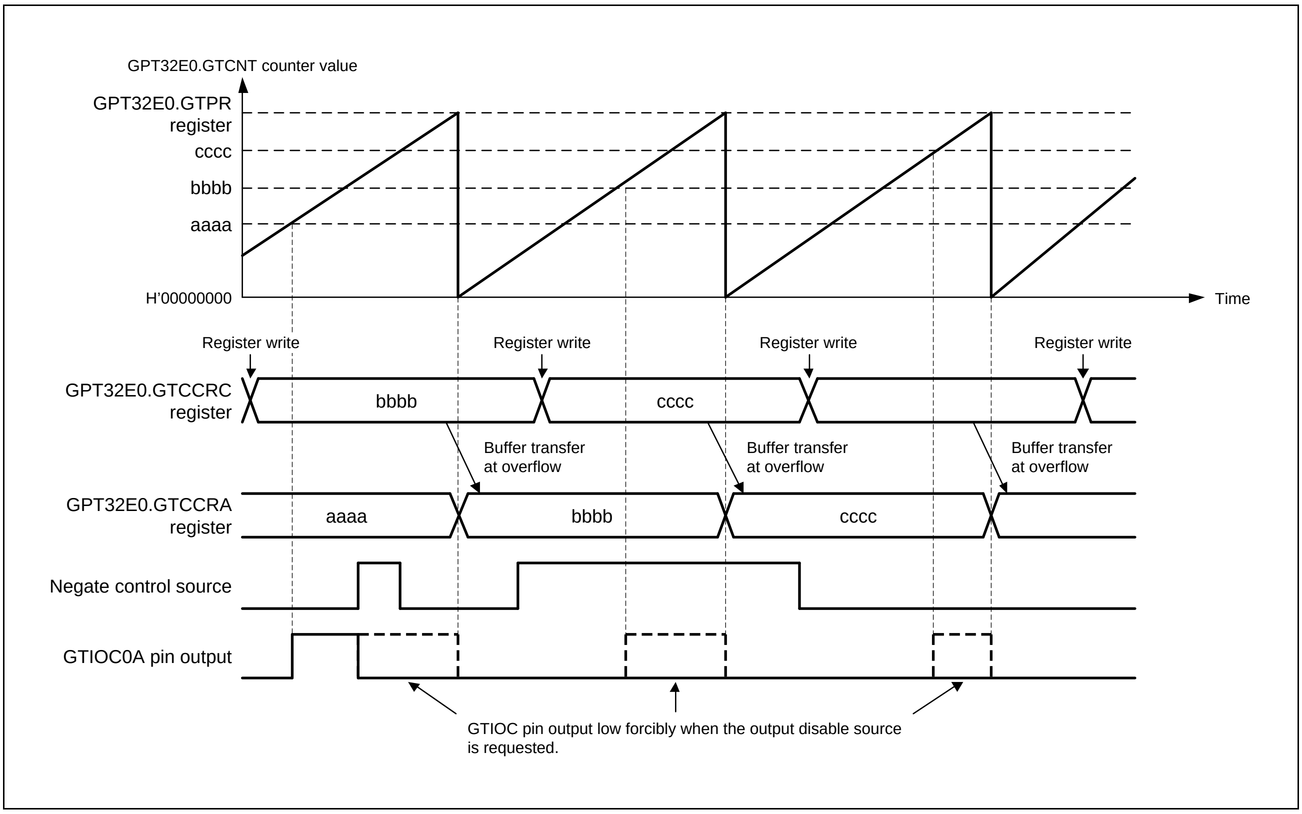


Figure 18.90 Example of GTIOC pin output disable control operation with saw-wave up-counting, buffer operation, active level 1, high output at GTCCRA compare match, low output at cycle end, and low output at output disable

18.7.4 Output Protection Function for GTIOC Pin Output

In preparation for incorrect settings of the GTCCRA register (settings outside the range of 0 < GTCCRA < GTPR), the output protection function for the GTIOC pin output (disabling function) is activated when the automatic dead time setting (GTDTCR.TDE = 1) is made in triangle-wave mode. The status of the output protection function can be read from GTSOS.SOS[1:0].

Figure 18.91 shows the output protection function state transition.

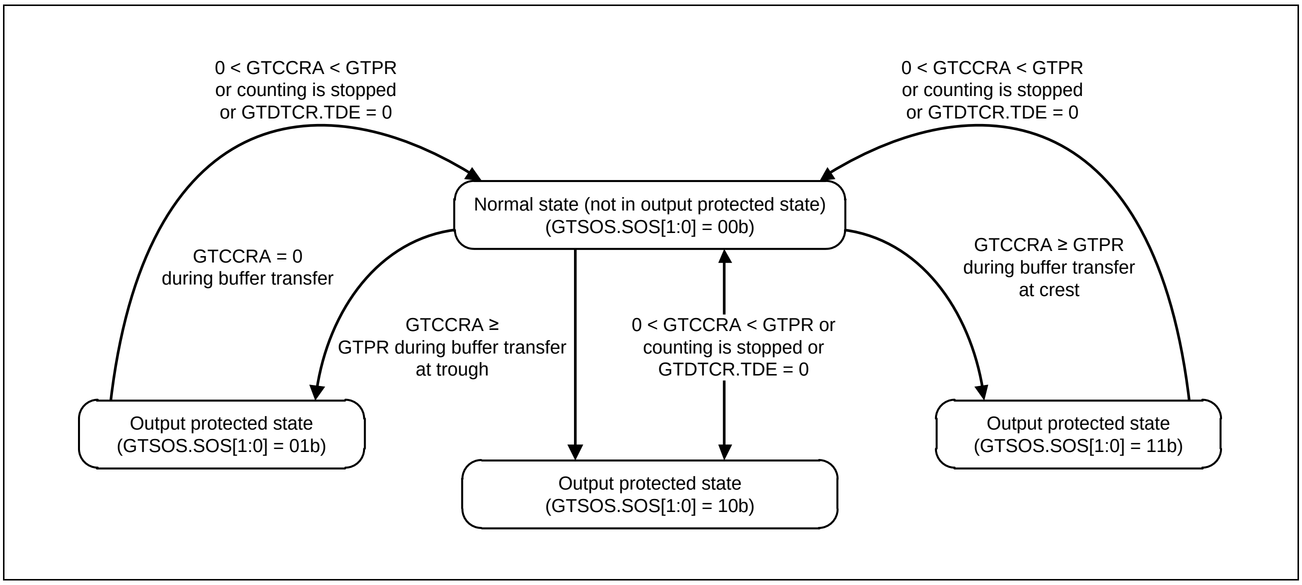
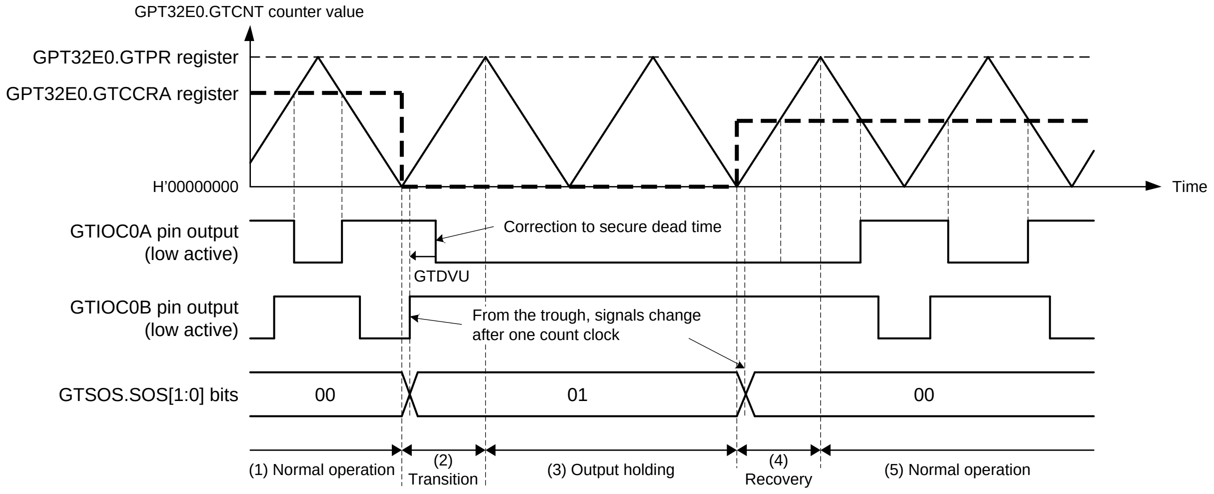


Figure 18.91 Output protection function

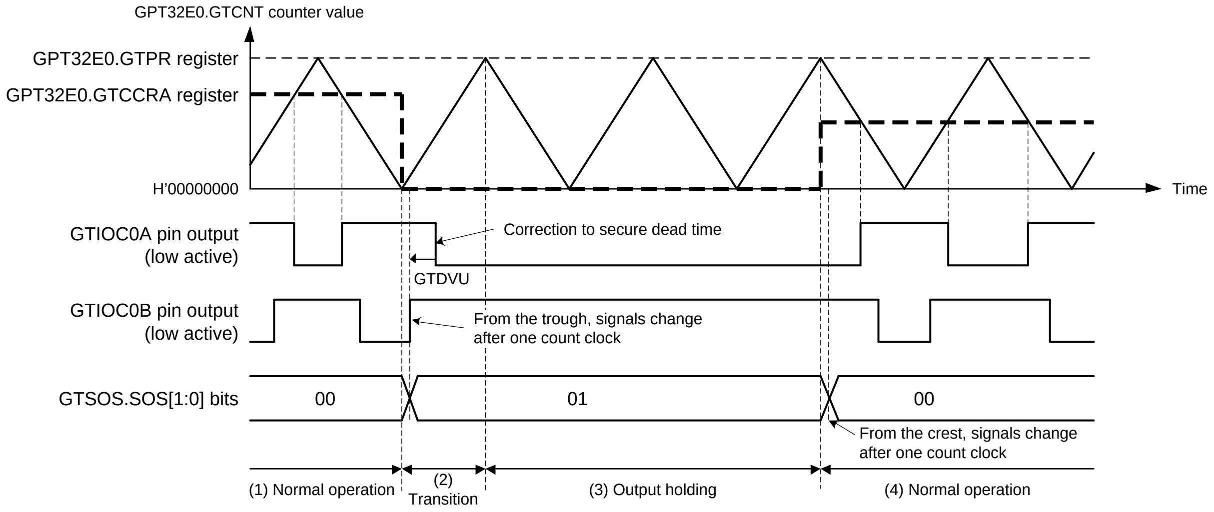
18.7.4.1 Output Protection Function When the GTCCRA Register is Set to 0 during Buffer Transfer

Figure 18.92 and **Figure 18.93** show examples of output protection function operation when the GTCCRA register is set to 0 during buffer transfer at troughs, and **Figure 18.94** and **Figure 18.95** show examples when the GTCCRA register is set to 0 during buffer transfer at crests.



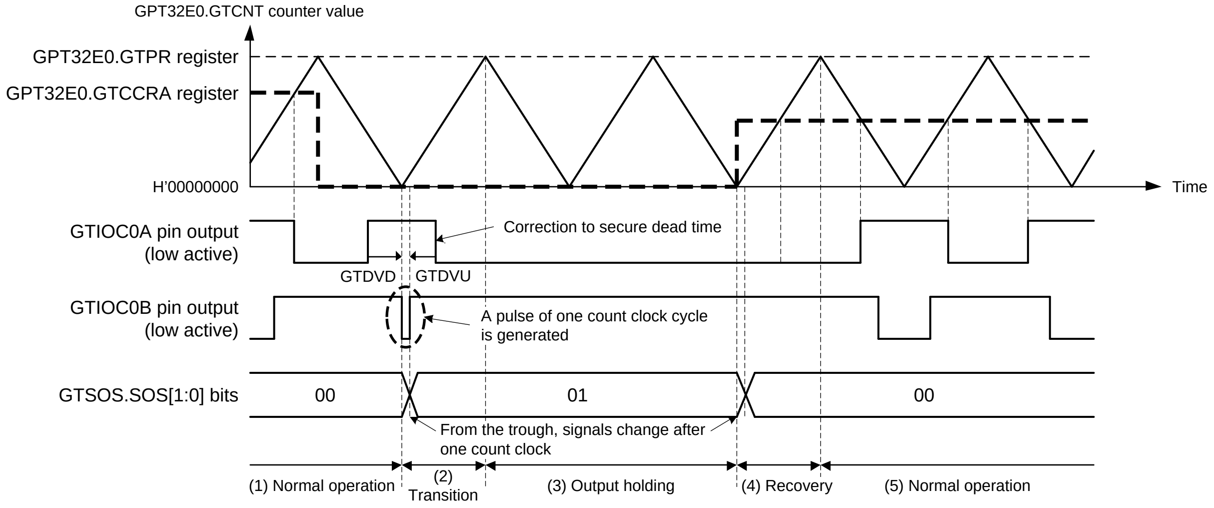
(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Transition period:	When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.
(3) Output holding period:	The output is held while GTCCRA register = 0.
(4) Recovery period:	When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next crest.
(5) Normal operation period:	Toggle output due to a compare match isperformed normally.

Figure 18.92 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low



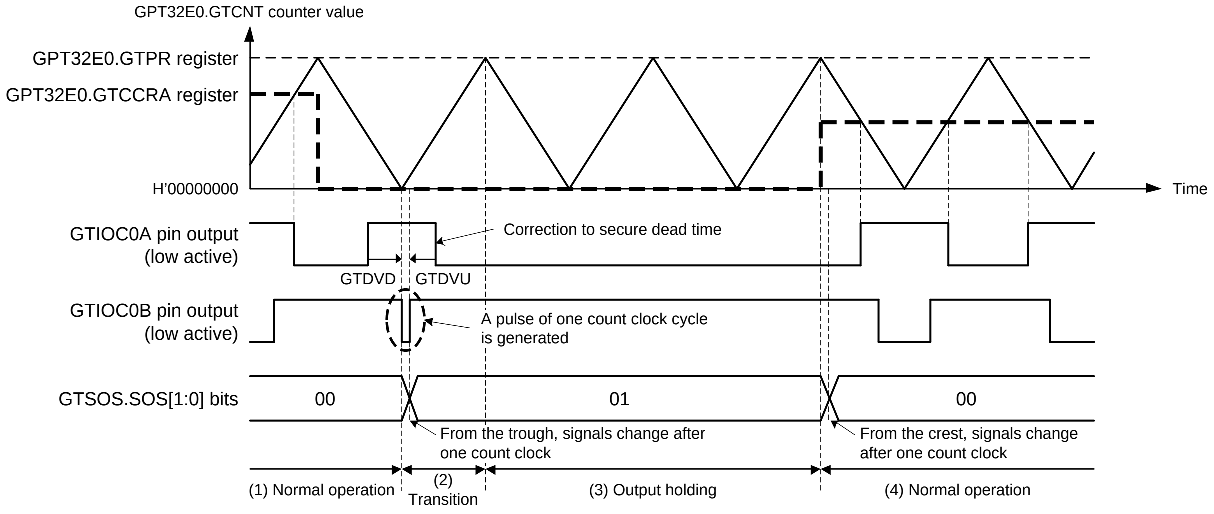
(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Transition period:	When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.
(3) Output holding period:	The output is held while GTCCRA register = 0.
(4) Normal operation period:	When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.93 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low



(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Transition period:	When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.
(3) Output holding period:	The output is held while GTCCRA register = 0.
(4) Recovery period:	When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next crest.
(5) Normal operation period:	Toggle output due to a compare match is performed normally.

Figure 18.94 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

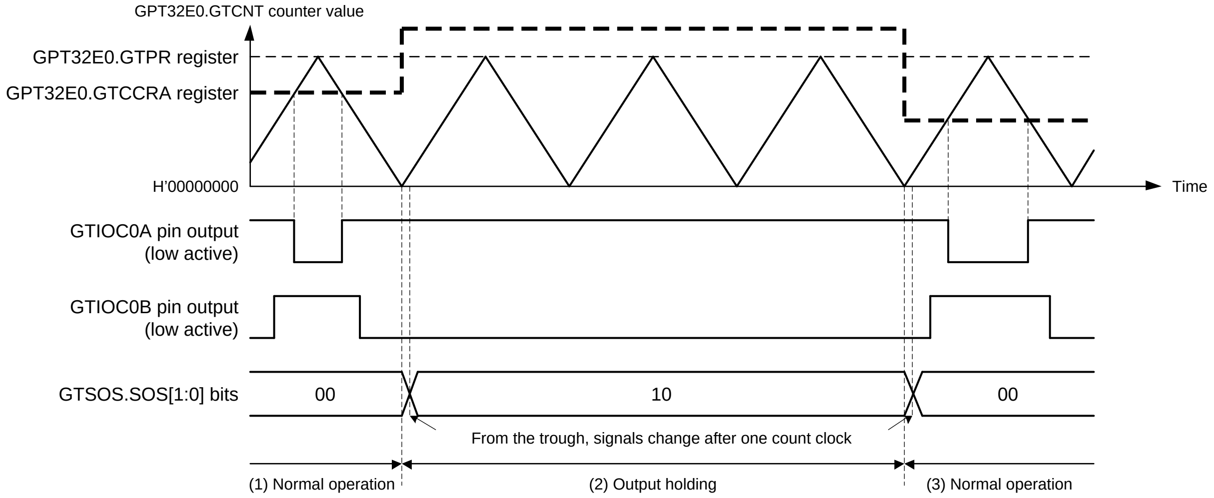


(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Transition period:	When GTCCRA register = 0 is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 01b, and toggle output is continued up to the next crest.
(3) Output holding period:	The output is held while GTCCRA register = 0.
(4) Normal operation period:	When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.95 Example of output protection operation when GTCCRA is set to 0 during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low

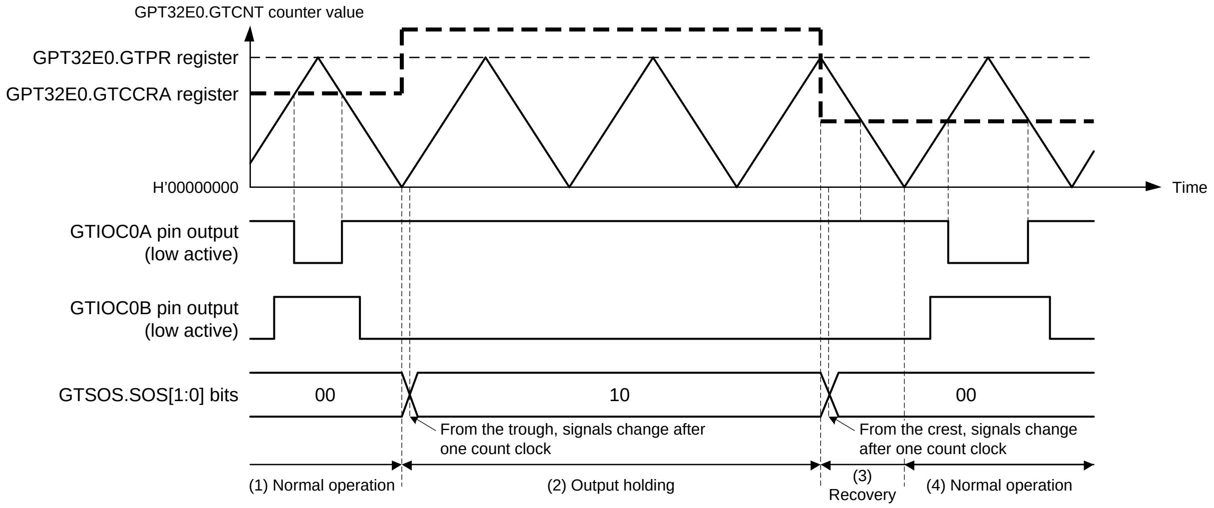
18.7.4.2 Output Protection Function When GTCCRA ≥ GTPR is Set during Buffer Transfer at Troughs

Figure 18.96 and **Figure 18.97** show examples of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs.



(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Output holding:	When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register.
(3) Normal operation period:	When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.96 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

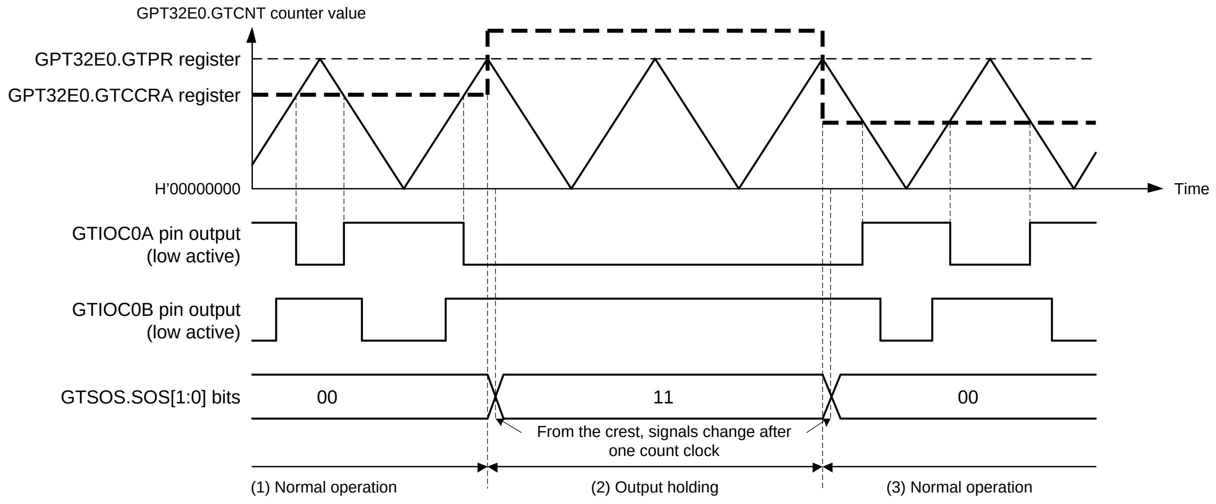


(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Output holding period:	When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register.
(3) Recovery period:	When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and holding output is maintained up to the next trough.
(4) Normal operation period:	Toggle output due to a compare match is performed normally.

Figure 18.97 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active-low

18.7.4.3 Output Protection Function When GTCCRA ≥ GTPR is Set during Buffer Transfer at Crests

Figure 18.98 and **Figure 18.99** show examples of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at crests.

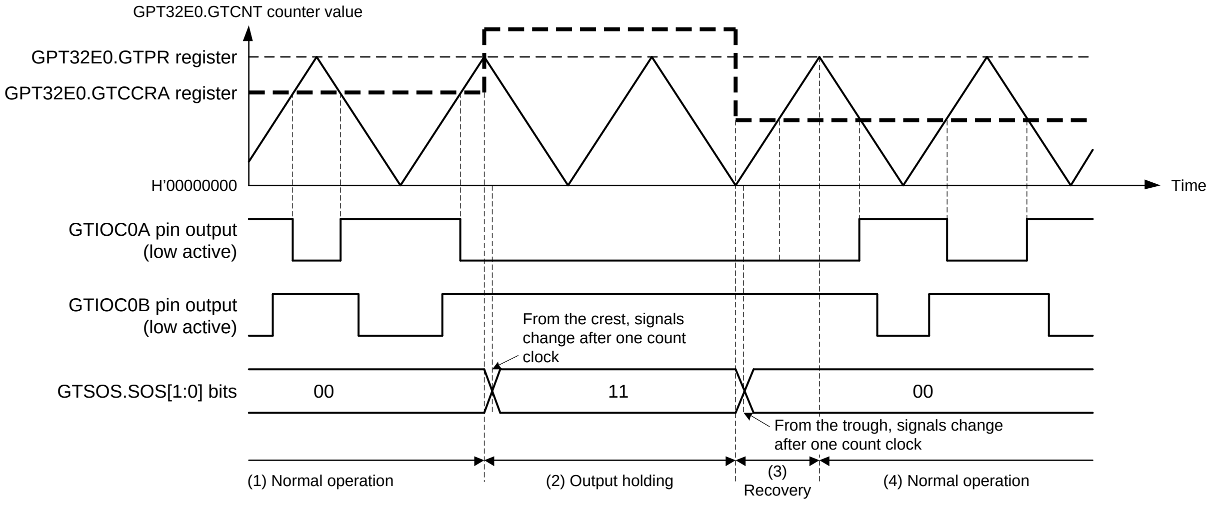


(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Output holding period:: When GTCCRA register ≥ GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 11b, and the output is held while GTCCRA register ≥ GTPR register.

(3) Normal operation period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] bits = 00b, and toggle output due to a compare match is performed normally.

Figure 18.98 Example of output protection operation when GTCCRA ≥ GTPR is set during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at crests, and active- low



(1) Normal operation period: Toggle output due to a compare match is performed normally.

(2) Output holding period: When GTCCRA ≥ GTPR is detected in the crest, the output protection status changes to GTSOS.SOS[1:0] = 11b, and the output is held while GTCCRA ≥ GTPR.

(3) Recovery period: When 0 < GTCCRA < GTPR is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] = 00b, and holding output is maintained up to the next crest.

(4) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 18.99 Example of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at crests, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

18.7.4.4 Restricted Specification of Output Protection Function

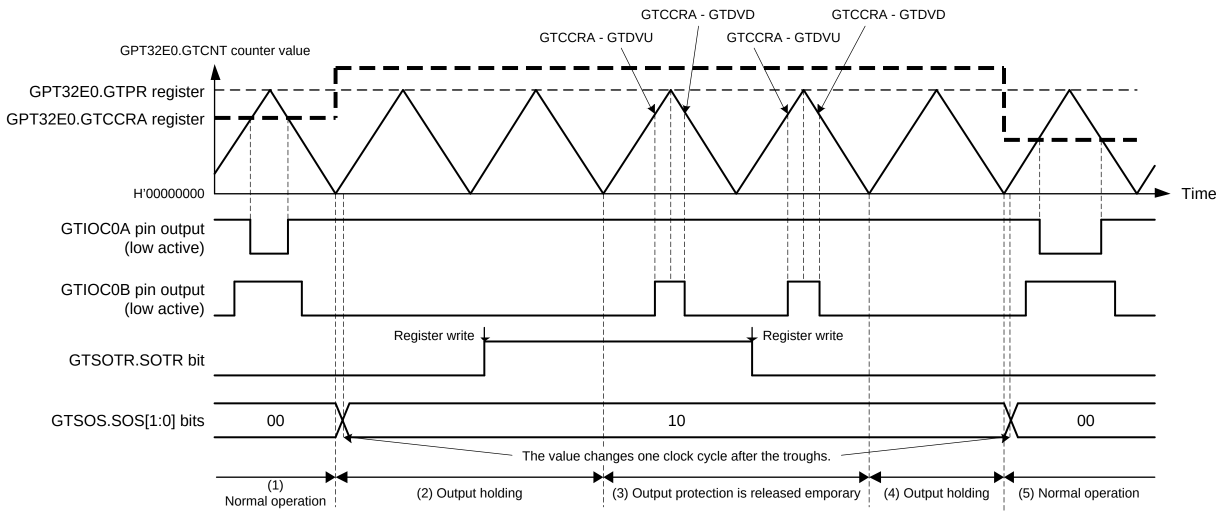
The output protection function deactivates the level of one of the positive and negative outputs, even if an incorrect value is set in the GTCCRA register during counting (a setting outside the range of 0 < GTCCRA < GTPR). However, it will not work properly unless the following conditions are satisfied.

- At count start: The value of the GTCCRA register must be set within the range of (0 < GTCCRA < GTPR).
- During buffer transfer at crest: GTCCRA - GTDVD < GTPR - 1 has to be satisfied..
- During buffer transfer at trough: If GTCCRA ≥ GTPR, GTCCRA - GTDVU > 1 has to be satisfied.

18.7.4.5 Temporary cancellation of Output Protection Function

When the GTSOTR.SOTR bit is set to 1 with GTSOS.SOS[1:0] bits equal to 10b (showing output protection state by GTCCRA ≥ GTPR during buffer transfer at troughs), the output protection function for GTIOCB pin is temporarily canceled. GTSOS.SOS[1:0] bits retain the value of 10b even when the output protection function is canceled. When the SOTR bit is set to 0, the output protection function for GTIOCB pin resumes.

Figure 18.100 shows examples of temporary cancellation of output protection function operation when the GTCCRA ≥ GTPR is set during buffer transfer at troughs.



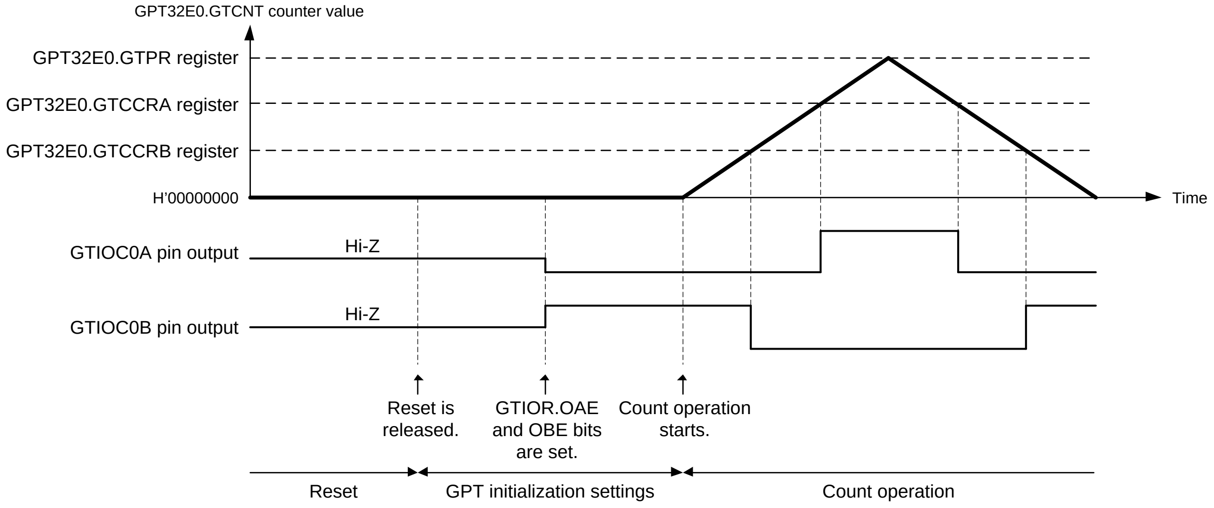
(1) Normal operation period:	Toggle output due to a compare match is performed normally.
(2) Output holding period:	When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 10b, and the output is held while GTCCRA register ≥ GTPR register. If GTSOTR.SOTR is set to 1 during the output holding period, the GTIOCB output is holding until the first trough after the setting.
(3) Output protection releasing period:	After GTSOTR.SOTR is set to 1, the GTIOCB output is performed normally at the first trough after the setting. If GTSOTR.SOTR is set to 0 during the output protection releasing period, the output is holding at the first trough after the setting.
(4) Output holding period:	Until 0 < GTCCRA register < GTPR register is detected in the trough, the output is holding. When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS.SOS[1:0] bits = 00b.
(5) Normal operation period:	Toggle output due to a compare match is performed normally.

Figure 18.100 Example of temporary cancellation of output protection function operation when GTCCRA ≥ GTPR is set during buffer transfer at troughs, with 0 < GTCCRA < GTPR restored during buffer transfer at troughs, and active-low

18.8 Initialization Method of Output Pins

18.8.1 Pin Settings after Reset

The GPT registers are initialized at a reset. Start counting after selecting the port mode, setting GTIOR.OAE and GTIOR.OBE bits, and outputting the GPT function to external pins.



[Setting examples]

GTIOR.GTIOA[4:0] bits: Initial output of low, output retained at cycle end, output toggled at compare match

GTIOR.GTIOB[4:0] bits: Initial output of high, output retained at cycle end, output toggled at compare match

Figure 18.101 Example of pin settings after reset

18.8.2 Pin Initialization Caused by Error during Operation

If an error occurs during GPT operation, the following four types of pin processing can be performed before pin initialization:

- Set the OAHLD and OBHLD bits in GTIOR to 1 and retain the outputs at count stop.
- Set the OAHLD and OBHLD bits in GTIOR to 0, specify arbitrary output values in OADFLT and OBDFLT in GTIOR, and output the arbitrary values on count stop.
- Set the pin to output an arbitrary value as a general output port by setting the PMn, Pn, and PMCn registers of the I/O port in advance. Set the OAE and OBE bits in GTIOR to 0 and the control bit associated with the pin in the PMCn to 0, to allow arbitrary values to be output from the pin set as a general output port when an error occurs.
- Drive the output to a high impedance state using the POEG function.

When the automatic dead time setting is made, clear the GTDTCR.TDE bit to 0 after counting stops. When counting stops, only the values of registers that are changed by a GPT external source change. If counting resumes, operation continues from where it stops. If counting stops, registers must be initialized before counting starts.

18.9 Usage Notes

18.9.1 GTCCRn Settings during Compare Match Operation (n = A to F)

(1) When automatic dead time setting is made in triangle-wave PWM mode

The GTCCRA register must satisfy the following conditions: GTDVU < GTCCRA, GTDVD < GTCCRA, and GTCCRA < GTPR.

When the setting of GTCCRA = 0 or GTCCRA ≥ GTPR is made during count operation, the output protection function is activated.

However, it will not work properly unless the following conditions are satisfied.

- At count start: The value of the GTCCRA register must be set within the range of (0 < GTCCRA < GTPR).
- During buffer transfer at crest: GTCCRA - GTDVD < GTPR - 1 has to be satisfied..
- During buffer transfer at through: If GTCCRA ≥ GTPR, GTCCRA - GTDVU > 1 has to be satisfied.

For details, see **Section 18.7.4, Output Protection Function for GTIOC Pin Output**.

(2) When automatic dead time setting is not made in triangle-wave PWM mode

The GTCCRA register must be set within the range of 0 < GTCCRA < GTPR. If GTCCRA = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRA = 0 or GTCCRA = GTPR is satisfied. When GTCCRA > GTPR, no compare match occurs.

Similarly, GTCCRB must be set within the range of 0 < GTCCRB < GTPR. If GTCCRB = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRB = 0 or GTCCRB = GTPR is satisfied. When GTCCRB > GTPR, no compare match occurs.

(3) When automatic dead time setting is made in saw-wave one-shot pulse mode

The GTCCRC and GTCCRD registers must be set to satisfy the following constraints. If the constraints are not satisfied, correct output waveforms with secured dead time might not be obtained.

- In up-counting: GTCCRC < GTCCRD, GTCCRC > GTDVU, GTCCRD < GTPR - GTDVD
- In down-counting: GTCCRC > GTCCRD, GTCCRC < GTPR – GTDVU, GTCCRD > GTDVD

(4) When automatic dead time setting is not made in saw-wave one-shot pulse mode

The GTCCRC and GTCCRD registers must be set to satisfy the following constraints. If the constraints are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRC < GTCCRD < GTPR
- In down-counting: GTPR > GTCCRC > GTCCRD > 0

Similarly, GTCCRE and GTCCRF must be set to satisfy the following constraints. If the constraints are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRE < GTCCRF < GTPR
- In down-counting: GTPR > GTCCRE > GTCCRF > 0

(5) In saw-wave PWM mode

The GTCCRA register must be set with the range of 0 < GTCCRA < GTPR. If GTCCRA = 0 or GTCCRA = GTPR is set, a compare match occurs within the cycle only when GTCCRA = 0 or GTCCRA = GTPR is satisfied. If GTCCRA > GTPR is set, no compare match occurs.

Similarly, GTCCRB must be set with the range of 0 < GTCCRB < GTPR. If GTCCRB = 0 or GTCCRB = GTPR is set, a compare match occurs within the cycle only when GTCCRB = 0 or GTCCRB = GTPR is satisfied. If GTCCRB > GTPR is set, no compare match occurs.

18.9.2 Setting Range for the GTCNT Counter

The GTCNT counter register must be set with the range of 0 ≤ GTCNT ≤ GTPR.

18.9.3 Starting and Stopping the GTCNT Counter

The control timing of starting and stopping the GTCNT counter by the GTCR.CST bit synchronizes the count clock that is selected in GTCR.TPCS[2:0]. When GTCR.CST is updated, the GTCNT counter starts/stops after a count clock that is selected in GTCR.TPCS[2:0]. Therefore, an event generated before the GTCNT counter actually starts is ignored.

Note, however, that an event might be accepted or an interrupt might occur after the GTCR.CST bit is set to 0.

18.9.4 Priority On Conflicts

(1) GTCNT register

Table 18.20 shows a priority order of events updating GTCNT register.

Table 18.20 Priority order of sources updating GTCNT

Source updating GTCNT	Priority order
Writing by CPU (Writing to GTCNT/GTCLR)	High
Clear by hardware sources set in GTCSR	↑
Count up or down by hardware sources set in GTUPSR/GTDNSR	
Count operation	Low

If up-counting and down-counting by hardware sources occur at the same time, the GTCNT counter value does not change. When there is a conflict between updating the GTCNT register and reading by the CPU, pre-update data is read.

(2) GTCR.CST bit

When there is a conflict between starting/stopping by hardware sources set in the GTSSR/GTPSR registers and writing by the CPU (writing to GTCR/GTSTR/GTSTP registers), writing by CPU has a priority over starting/stopping by hardware sources.

When there is a conflict between starting by hardware sources set in the GTSSR register and stopping by hardware sources set in GTPSR register, the GTCR.CST bit value does not change. Where there is a conflict between updating the GTCR.CST bit and reading by the CPU, pre-update data is read.

(3) GTCCRn registers (n = A to F)

When there is a conflict between input capture/buffer transfer operation and writing to GTCCRn registers, writing to GTCCRn registers has a priority over input capture/buffer transfer operation. When there is a conflict between input capture and writing to the counter register by the CPU or updating the counter register by hardware sources, the pre-update counter value is captured. Where there is a conflict between updating the GTCCRn registers and reading by the CPU, pre-update data is read.

(4) GTPR registers

When there is a conflict between buffer transfer operation and writing to the GTPR register, writing to GTPR register has a priority over buffer transfer operation. When there is a conflict between updating GTPR register and reading by the CPU, pre-update data is read.

(5) GTADTRn registers (n = A, B)

When there is a conflict between buffer transfer operation and writing to the GTADTRn registers, writing to the GTADTRn registers has priority over buffer transfer operation. Where there is a conflict between updating GTADTRn registers and reading by the CPU, pre-update data is read.

(6) GTDVn registers (n = U, D)

When there is a conflict between buffer transfer operation and writing to GTDVn registers, writing to GTDVn registers has priority over buffer transfer operation. When there is a conflict between updating GTDVn registers and reading by the CPU, pre-update data is read.

19. Port Output Enable for GPT (POEG)

19.1 Overview

The output pins of the general PWM timer (GPT) can be disabled by using the port output enabling function for the GPT (POEG). Specifically, either of the following ways can be used.

- Input level detection of the GTETRGA to GTETRGD pins
- Output-disable request from the GPT
- Register settings

The GTETRGA to GTETRGD pins can also be used as GPT external trigger input pins.

Table 19.1 lists the POEG specifications, **Figure 19.1** shows the block diagram, and **Table 19.2** lists the input pins.

Table 19.1 POEG Specifications

Parameter	Specifications
Output-disable control through input level detection	GPT output pins can be disabled when a GTETRGA to GTETRGD rising edge or high level is sampled after polarity and filter selection
Output-disable request from the GPT	• When the GTIOCA pin and the GTIOCB pin are driven to an active level simultaneously, the GPT generates an output-disable request to the POEG. Through reception of these requests, the POEG can control whether the GTIOCA and GTIOCB pins are output-disabled • GPT output pins can be set to be disabled when the GPT output pins detect a dead time error or short circuit detection between the output terminals
Output-disable control by software (registers)	GPT output pins can be disabled by modifying the register settings
Interrupt	• Allows output-disable control by input level detection • Allows output-disable requests from the GPT
External trigger output to the GPT (count start, count stop, count clear, up-count, down-count, or input capture function)	GTETRGA to GTETRGD signals can be output to the GPT after polarity and filter selection
Noise filtering	• Three times sampling for every P0ϕ1, P0ϕ8, P0ϕ32, or P0ϕ128 can be set for any of the input pins GTETRGA to GTETRGD • Positive or negative polarity can be selected for any of the input pins, GTETRGA to GTETRGD • Signal state after polarity and filter selection can be monitored

GTETRGn (n = A to D) in the subsequent descriptions indicates the GTETRGA to GTETRGD pins. The signal input from each of the pins corresponds to the POEGGn register for which n matches the n of the pin.

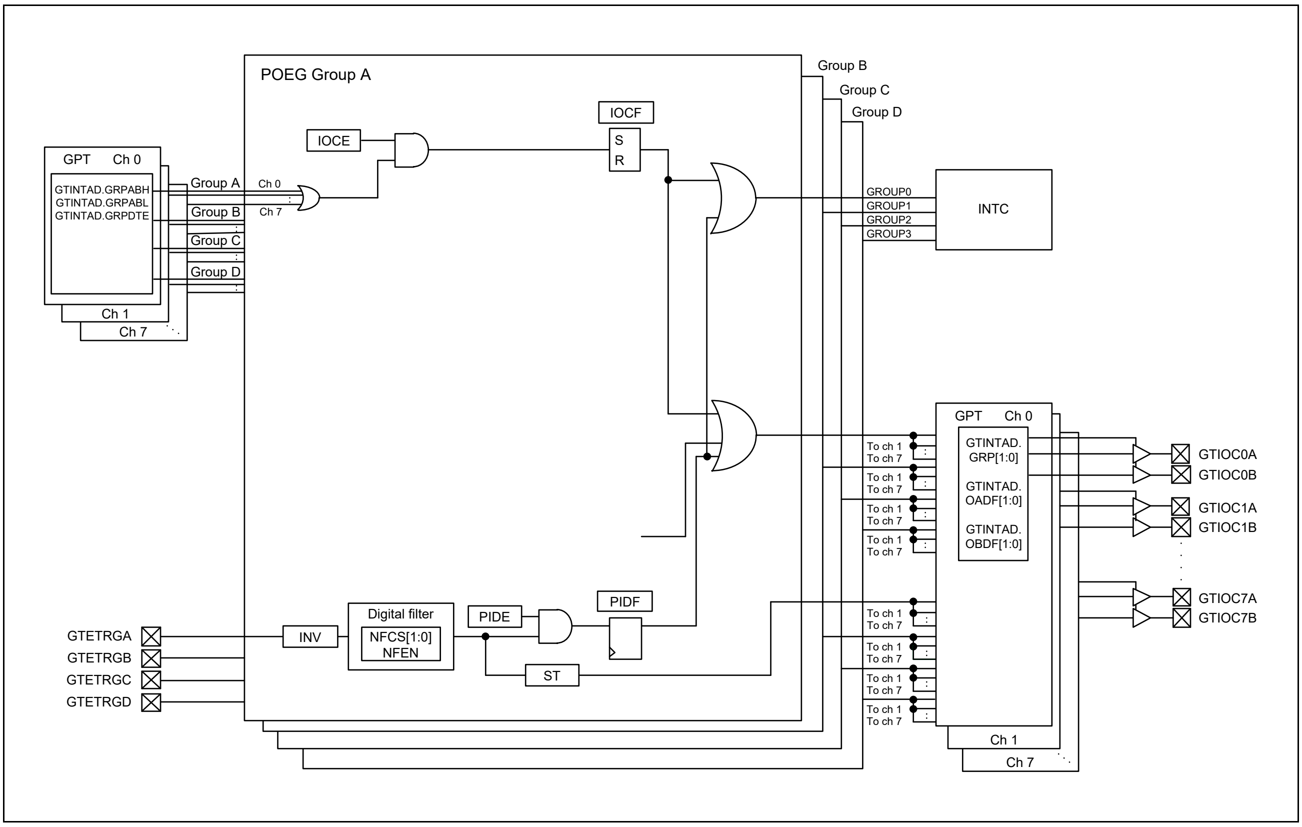


Figure 19.1 POEG Block Diagram

Table 19.2 POEG Input Pins

Pin Name	I/O	Description
GTETRGA	Input	GPT output pin output-disable request signal and GPT external trigger input pin A
GTETRGB	Input	GPT output pin output-disable request signal and GPT external trigger input pin B
GTETRGC	Input	GPT output pin output-disable request signal and GPT external trigger input pin C
GTETRGD	Input	GPT output pin output-disable request signal and GPT external trigger input pin D

19.2 Register Descriptions

Table 19.3 to **Table 19.5** shows the register configuration.

Table 19.3 Register Configuration 1 (Cortex-A55 Address Space)

Register Name	Abbreviation	Address	Access Size
POEG group A setting register	POEGGA	H'0_1004_8800	32
POEG group B setting register	POEGGB	H'0_1004_8C00	32
POEG group C setting register	POEGGC	H'0_1004_9000	32
POEG group D setting register	POEGGD	H'0_1004_9400	32

Table 19.4 Register Configuration 2 (Cortex-M33 Address Space Non-Secure)

Register Name	Abbreviation	Address	Access Size
POEG group A setting register	POEGGA	H'4004_8800	32
POEG group B setting register	POEGGB	H'4004_8C00	32
POEG group C setting register	POEGGC	H'4004_9000	32
POEG group D setting register	POEGGD	H'4004_9400	32

Table 19.5 Register Configuration 3 (Cortex-M33 Address Space Secure)

Register Name	Abbreviation	Address	Access Size
POEG group A setting register	POEGGA	H'5004_8800	32
POEG group B setting register	POEGGB	H'5004_8C00	32
POEG group C setting register	POEGGC	H'5004_9000	32
POEG group D setting register	POEGGD	H'5004_9400	32

19.2.1 POEG Group n Setting Register (POEGGn) (n = A to D)

Bit	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NFCS[1:0]		NFEN	INV	—	—	—	—	—	—	—	—	—	—	—	ST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	IOCE	PIDE	SSF	—	IOCF	PIDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*2	R/W*2	R/W	R/W	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
b0	PIDF	0	R(/W) *1	Port Input Detection Flag 0: No output-disable request from the GTETRGn pin occurred 1: Output-disable request from the GTETRGn pin occurred.
b1	IOCF	0	R(/W) *1	Detection Flag for GPT Output-Disable Request 0: No output-disable request from GPT disable request occurred 1: Output-disable request from GPT disable request occurred.
b2	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b3	SSF	0	R/W	Software Stop Flag 0: No output-disable request from software occurred 1: Output-disable request from software occurred.
b4	PIDE	0	R/W*2	Port Input Detection Enable 0: Disable output-disable requests from the GTETRGn pins 1: Enable output-disable requests from the GTETRGn pins.
b5	IOCE	0	R/W*2	Enable for GPT Output-Disable Request 0: Disable output-disable requests from GPT disable request 1: Enable output-disable requests from GPT disable request.
b15 to b6	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b16	ST	0	R	GTETRGn Input Status Flag 0: GTETRGn input after filtering was 0 1: GTETRGn input after filtering was 1.
b27 to b17	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b28	INV	0	R/W	GTETRGn Input Reverse 0: Input GTETRGn as-is 1: Input GTETRGn in reverse.
b29	NFEN	0	R/W	Noise Filter Enable 0: Disable noise filtering 1: Enable noise filtering.

Bit	Bit Name	Initial Value	R/W	Description
b31, b30	NFCS[1:0]	All 0	R/W	Noise Filter Clock Select b1 b0 0 0: Sample GTETRGn pin input level three times every P0ϕ 0 1: Sample GTETRGn pin input level three times every P0ϕ8 1 0: Sample GTETRGn pin input level three times every P0ϕ32 1 1: Sample GTETRGn pin input level three times every P0ϕ128.

Note 1. Only 0 can be written, to clear the flag.

Note 2. Can be modified only once after a reset.

The POEGGA to POEGGD registers control the output-disable state of the GPT pins, interrupts, and the external trigger input to the GPT. In the descriptions, POEGGn represents all of the POEGGA to POEGGD registers.

19.3 Output-Disable Control Operation

The output of the GTIOCxA and GTIOCxB pins can be disabled when any of the following conditions are satisfied.

- Input level or edge detection of the GTETRGn pins

  When POEGGn.PIDE is 1, the POEGGn.PIDF flag is set to 1.

- Output-disable request from the GPT

  When POEGGn.IOCE is 1, the POEGGn.IOCF flag is set to 1.
  The output-disable requests enabled by GRPDTE, GRPABH, and GRPABL bits of the GTINTAD register in the GPT are applied to the group selected by GRP[1:0] bits of the GTINTAD register.

- SSF bit setting

  When POEGGn.SSF is set to 1.

The state of the GTIOCxA and the GTIOCxB pins when the output is disabled is controlled by the GPT module.

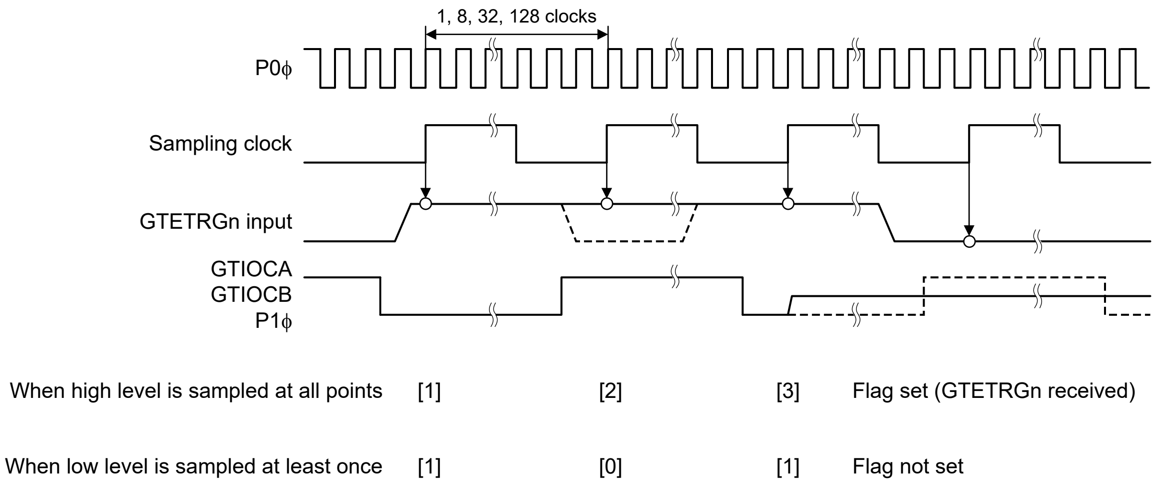
For details, see the descriptions of the GTINTAD.GRP[1:0], GTIOR.OADF[1:0], and GTIOR.OBDF[1:0] bits of the general PWM timer (GPT).

19.3.1 Pin Input Level Detection Operation

If the input conditions set in POEGGn.PIDE, POEGGn.NFCS[1:0], POEGGn.NFEN, and POEGGn.INV occur on the GTETRGn pins, the GPT output pins are output-disabled.

Digital Filter

Figure 19.2 shows high-level detection by the digital filter. When a high level associated with the POEGGn.INV polarity setting is detected three times consecutively with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, the detected level is recognized as high, and the GPT output pins are output-disabled. If even one low level is detected during this interval, the detected level is not recognized as high. In addition, in an interval where the sampling clock is not being output, changes of the levels on the GTETRGn pins are ignored.



Note: Individual channel output can be set in the GPT settings.
Low level sampling can be selected in POEGGn.INV.

Figure 19.2 Example of Digital Filter Operation

19.3.2 Output-Disable Requests from the GPT

For details on this operation, see the description of **18.7.3, GTIOC Pin Output Negate Control** in **Section 18, General PWM Timer (GPT)**.

19.3.3 Output-Disable Control Using Registers

The GPT output pins can be directly controlled by writing to the software stop flag, POEGGn.SSF.

19.3.4 Release from Output-Disable

To release the GPT output pins placed in the output-disable state, either return them to their initial state with a reset or clear all of the following:

- POEGGn.PIDF flag
- POEGGn.IOCF flag
- POEGGn.SSF flag.

Writing 0 to the POEGGn.PIDF flag is ignored (the flag is not cleared) if the external input pins, GTETRGn, are not disabled and the POEGGn.ST bit is not set to 0.

Writing 0 to the POEGGn.IOCF flag is valid (the flag is cleared) only if all of the GTST.DTEF, GTST.OABHF, and GTST.OABLF flags in the GPT are set to 0.

Figure 19.3 shows the release timing for output-disable. The output-disable is released at the beginning of the next count cycle of the GPT after the flag is cleared.

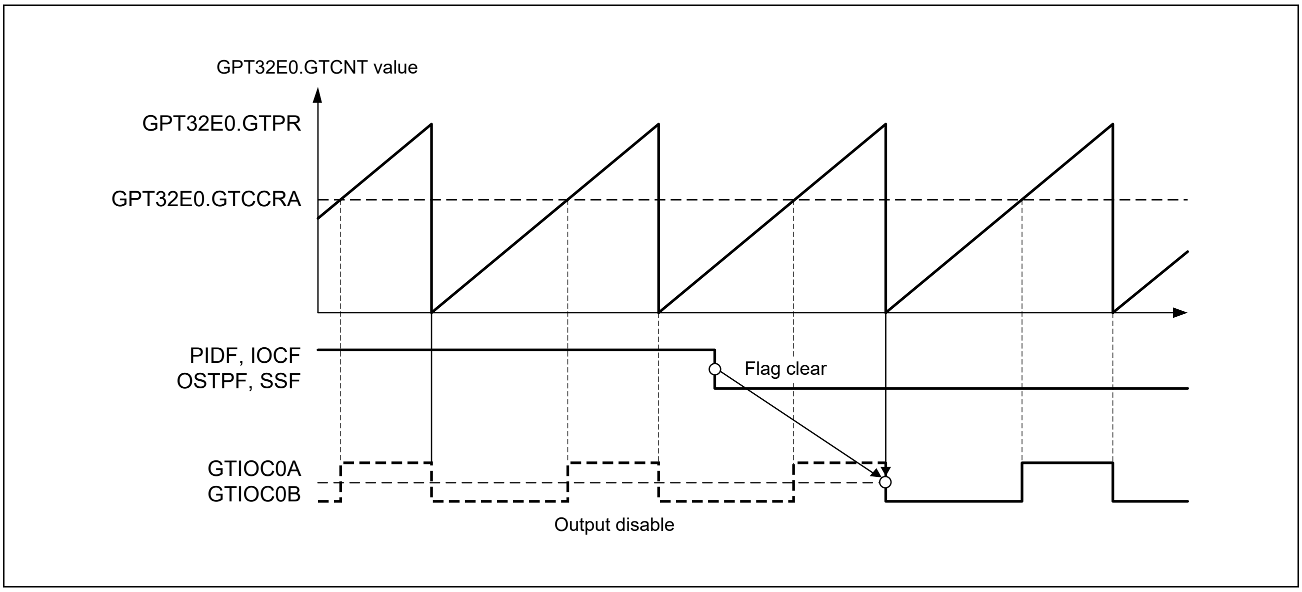


Figure 19.3 Output-Disable Release Timing for GPT Pin Outputs

19.4 Interrupt Sources

The POEG generates an interrupt request when triggered by these sources:

- Output-disable control by input level detection
- Output-disable request from the GPT
- Output-disable control by the registers.

Table 19.6 lists the conditions for interrupt requests.

Table 19.6 Interrupt Sources and Conditions

Interrupt Source	Symbol	Associated Flag	Trigger Conditions
POEG group A interrupt	GROUP0	POEGGA.IOCF	Output-disable request from a GPT disable request occurred
		POEGGA.PIDF	Output-disable request from the GTETRGA pin occurred
POEG group B interrupt	GROUP1	POEGGB.IOCF	Output-disable request from a GPT disable request occurred
		POEGGB.PIDF	Output-disable request from the GTETRGB pin occurred
POEG group C interrupt	GROUP2	POEGGC.IOCF	Output-disable request from a GPT disable request occurred
		POEGGC.PIDF	Output-disable request from the GTETRGC pin occurred
POEG group D interrupt	GROUP3	POEGGD.IOCF	Output-disable request from a GPT disable request occurred
		POEGGD.PIDF	Output-disable request from the GTETRGD pin occurred

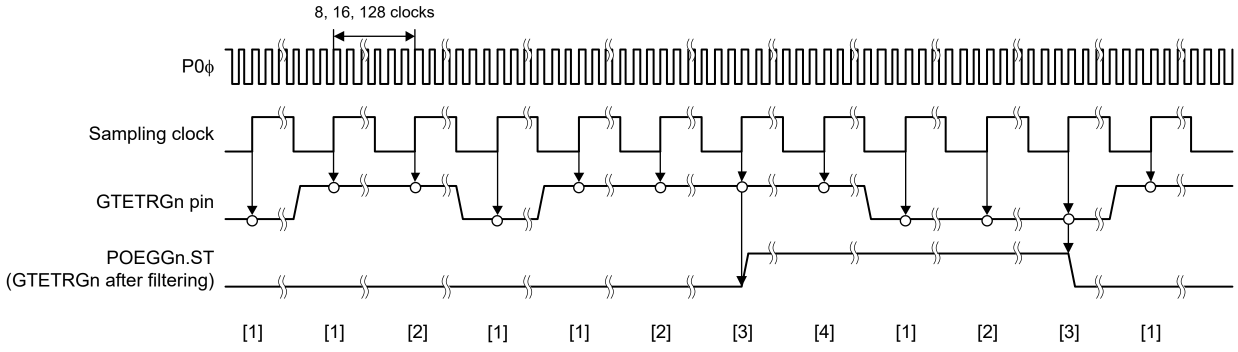
19.5 External Trigger Output to the GPT

The POEG outputs the GTETRGn signals to the GPT as the GPT operation trigger signal for the following:

- Count start
- Count stop
- Count clear
- Up-count
- Down-count
- Input capture.

For the POEGGn.INV polarity setting signal, when the same level is input three times continuously with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, that value is output. Set the control registers the same as for the input level detection operation described in **Section 19.3.1, Pin Input Level Detection Operation**. The state after filtering can be monitored in POEGGn.ST.

Figure 19.4 shows the output timing of an external trigger to the GPT.



Note: Each channel output can be set in the GPT settings.
Polarity can be reversed in POEGGn.INV.

Figure 19.4 Output Timing of External Trigger to GPT

19.6 Usage Notes

19.6.1 Specifying Pins Associated with the GPT

The POEG controls output-disable only when a pin is associated with the GPT in the PMCn and PFCm register settings of IO_TOP. When the pin is specified as a general I/O pin, the POEG does not perform output-disable control.

20. General Timer (GTM)

20.1 Functional Overview

The General timer has the following features.

- Two operating modes
  - Interval timer mode
  - Free-running comparison mode
- Choice between startup of DMA by compare match and generation of interrupt

20.1.1 Features of GTM

Channels

This product has the following number of channels of the General timer.

Table 20.1 Channels of General Timer

General Timer	
Number of Channels	3
Name	OSTMn

Note: n = 0, 1, 2

Meaning of n

Throughout this section, the individual channels of the General timer are identified by the index “n” (n = 0, 1, 2), for example OSTMnTO for the General timer n output register.

Register address

The register addresses of the General timer are given as offsets from the individual base addresses <OSTMn_base>. The register base addresses of each OSTMn are listed in the following table.

Table 20.2 Register Base Addresses

Base Address Name	Base Address
<OSTM0_base>	H’0_1280_1000 (H’4280_1000*1, H’5280_1000*2)
<OSTM1_base>	H’0_1280_1400 (H’4280_1400*1, H’5280_1400*2)
<OSTM2_base>	H’0_1280_1800 (H’4280_1800*1, H’5280_1800*2)

Note 1. Cortex-M33 Address Space Non-Secure

Note 2. Cortex-M33 Address Space Secure

Interrupts

The General timers can generate the following interrupt requests.

Table 20.3 OSTMn Interrupt Requests

OSTMn Signal	Function	Startup of Direct Memory Access Controller
OSTM0TINT	OSTM0 interrupt	✓
OSTM1TINT	OSTM1 interrupt	✓
OSTM2TINT	OSTM2 interrupt	✓

20.2 Registers

The General timer is controlled by the following registers.

20.2.1 Registers Overview

The list of OSTMn (n = 0, 1, 2) registers and the memory addresses are as follows.

For the base addresses, see **Table 20.2**.

For the actual addresses, the offset values indicated in the following table are added to the base addresses.

Register Name	Function	R/W	Reset Value	Access Unit (bit)			Address
				8	16	32	
OSTMnCMP	OSTM compare register	R/W	H’0000 0000	—	—	✓	<OSTMn_base> + H’00
OSTMnCNT	OSTM counter register	R	H’FFFF FFFF	—	—	✓	<OSTMn_base> + H’04
OSTMnTE	OSTM count enable status register	R	H’00	✓	—	—	<OSTMn_base> + H’10
OSTMnTS	OSTM count start trigger register	W	H’00	✓	—	—	<OSTMn_base> + H’14
OSTMnTT	OSTM count stop trigger register	W	H’00	✓	—	—	<OSTMn_base> + H’18
OSTMnCTL	OSTM control register	R/W	H’00	✓	—	—	<OSTMn_base> + H’20

20.2.2 Details of OSTM Registers

20.2.2.1 OSTMnCMP — OSTM Compare Register

Depending on the mode of operation, this register holds the start value for the down-counter or the value for comparison with that of the counter.

Access Size:	This register is readable/writable in 32-bit units.
Address(es):	<OSTMn_base>
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCMP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCMP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OSTMnCMP[31:0]	All 0	R/W	● In interval timer mode: start value of the down-counter ● In free-running comparison mode: value for comparison

20.2.2.2 OSTMnCNT — OSTM Counter Register

This register indicates the counter value of the timer.

Access Size: This register is readable in 32-bit units.

Address(es): <OSTMn_base> + H'4

Initial Value: The initial value depends on the operating mode of the General timer. Refer to **Table 20.4**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCNT[31:16]															
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCNT[15:0]															
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OSTMnCNT [31:0]	*1	R	32-bit counter value

Note 1. The initial value depends on the operating mode of the General timer. Refer to **Table 20.4**

The following table shows the correspondence between operating mode, counting direction and initial value. The initial value is the value read from the counter after a change to the operating mode.

Table 20.4 Correspondence between Operating Mode, Counting Direction and Initial Value

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Initial value
Interval timer mode	0*1	Down	H'FFFF FFFF
Free-running comparison mode	1	Up	H'0000 0000

Note 1. Value after reset

20.2.2.3 OSTMnTE — OSTM Count Enable Status Register

This register indicates whether the counter is enabled or disabled.

Access Size: This register is readable in 8-bit units.
Address(es): <OSTMn_base>+ H'10
Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMn TE
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTE	0	R	This bit indicates whether the counter is enabled or disabled. 0: Counter disabled 1: Counter enabled This bit is set to 1 in response to OSTMnTS.OSTMnTS being set to 1. This bit is reset to 0 in response to OSTMnTT.OSTMnTT being set to 1.

Note 1. When OSTMnTE = 0, the counter retains its value.
If the counter is restarted, it
- restarts counting down from the value in the OSTMnCMP register if it is in interval timer mode or
- restarts counting up from the counter value H'0000 0000 if it is in free running comparison mode.

Note 2. For debugging, the GTM counter can be paused by the counter stop request.

20.2.2.4 OSTMnTS — OSTM Count Start Trigger Register

This register starts the counter.

Access Size: This register is writable in 8-bit units. It is always read as H'00.
Address(es): <OSTMn_base>+ H'14
Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMn TS
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTS	0	W	This bit starts the counter. 0: This setting has no effect. 1: Starts the counter and sets OSTMnTE.OSTMnTE = 1. ● In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1. ● In free-running comparison mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1.

20.2.2.5 OSTMnTT — OSTM Count Stop Trigger Register

This register stops the counter.

Access Size: This register is writable in 8-bit units. It is always read as H'00.
Address(es): <OSTMn_base>+ H'18
Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMn TT
Initial Value								
R/W	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTT	0	W	Stops the counter. 0: This setting has no effect. 1: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

20.2.2.6 OSTMnCTL — OSTM Control Register

This register specifies the operating mode for the counter and controls enabling/disabling of OSTMnTINT interrupt requests when counting starts.

Access Size:	This register is readable/writable in 8-bit units. Writing to this register is only possible if the counter is disabled (OSTMnTOE.OSTMnTOE = 0).
Address(es):	<OSTMn_base>+ H'20
Initial Value:	H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OSTMn MD1	OSTMn MD0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	OSTMnMD1	0	R/W	Specifies the operating mode for the counter. 0: Interval timer mode 1: Free-running comparison mode
0	OSTMnMD0	0	R/W	Controls enabling/disabling of OSTMnTINT interrupt requests when counting starts. 0: Disables the interrupts when counting starts. 1: Enables the interrupts when counting starts.

20.3 Functional Description

Each General timer is a 32-bit timer/counter.

The settings for operating mode specify the direction of counting (up or down) and the generation of interrupt requests.

20.3.1 Block Diagram

The following block diagram shows the main components of GTM.

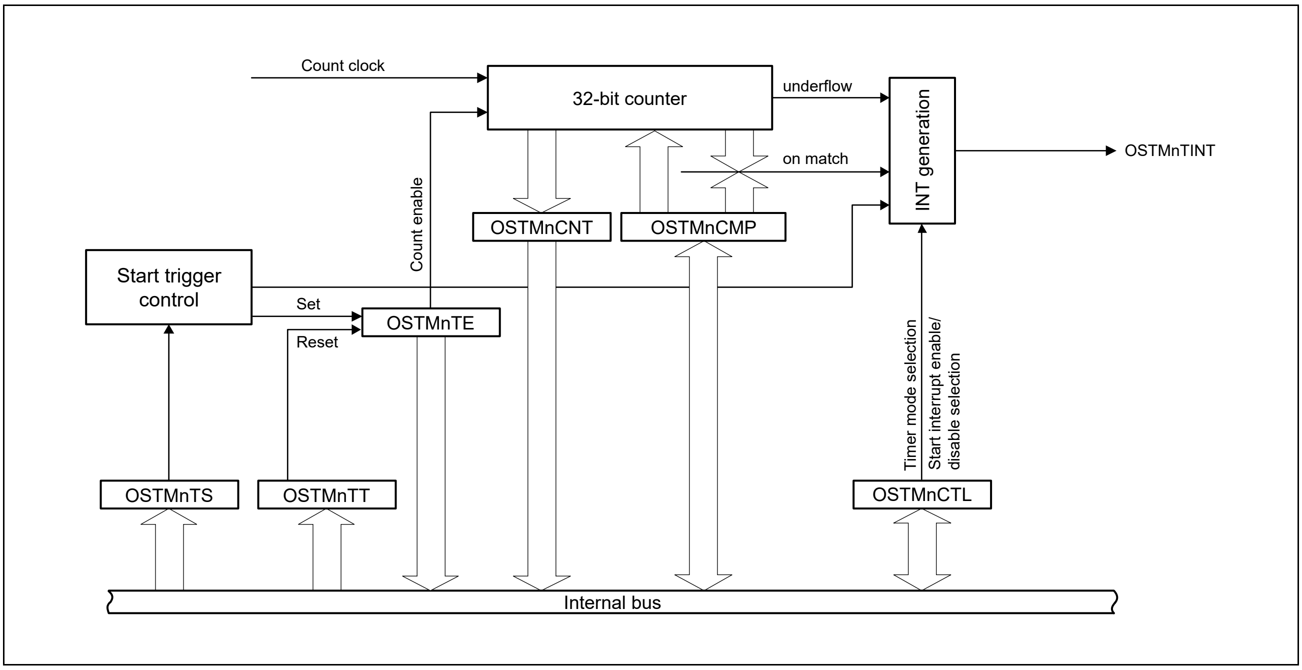


Figure 20.1 Block Diagram of GTM

20.3.2 Count Clock

The count clock of OSTMn is P0ϕ (OSTMn_PCLK).

20.3.3 Generation of Interrupt Request

An OSTMnTINT interrupt request is generated whenever the counter reaches H'0000 0000 (in interval timer mode) or matches the comparison value (in free-running comparison mode).

An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.

This operation is shown in the following figure.

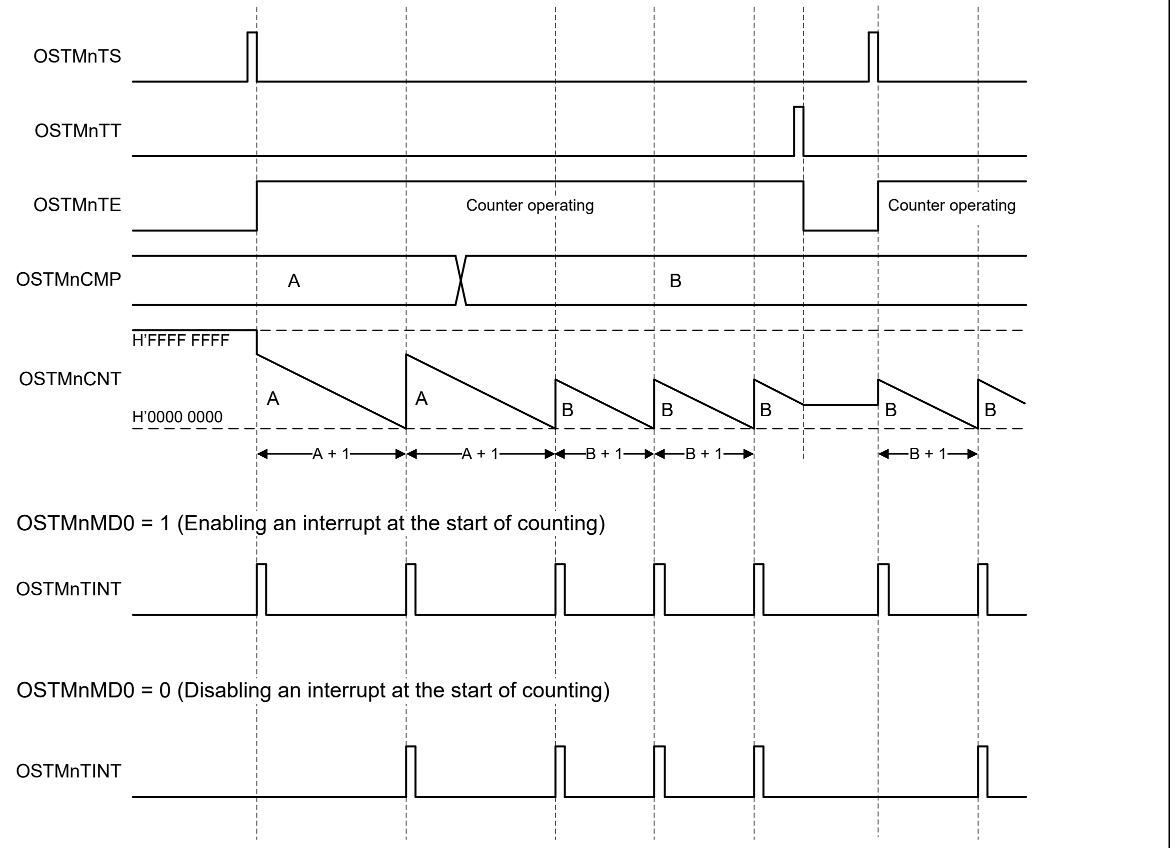


Figure 20.2 Generating an Interrupt when Counting Starts (in Interval Timer Mode)

20.3.4 Starting and Stopping the Timer

The General timer is started and stopped as follows.

Starting the timer

The timer is started in either of the following way:

- setting the OSTMnTS.OSTMnTSF bit to 1

Status bit OSTMnTE.OSTMnTE is set to 1.

The counter starts to count up or down in accord with the settings for operating mode.

Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.

This also clears the OSTMnTE.OSTMnTE status flag.

20.3.5 Interval Timer Mode

Select the interval timer mode when an General timer is to be used as a reference timer for generating interrupt requests at a fixed interval.

20.3.5.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An OSTMnTINT interrupt request is generated when the counter reaches H’0000 0000.

Select interval timer mode by setting OSTMnCTL.OSTMnMD1 = 0.

New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next H’0000 0000 is reached.

Cycles of OSTMnTINT output

The cycle of OSTMnTINT output is as follows.

- OSTMnTINT generation cycle = counter-clock cycle × (OSTMnCMP + 1)

The following figure shows the basic operation of OSTM when counter-start interrupts is enabled in interval timer mode.

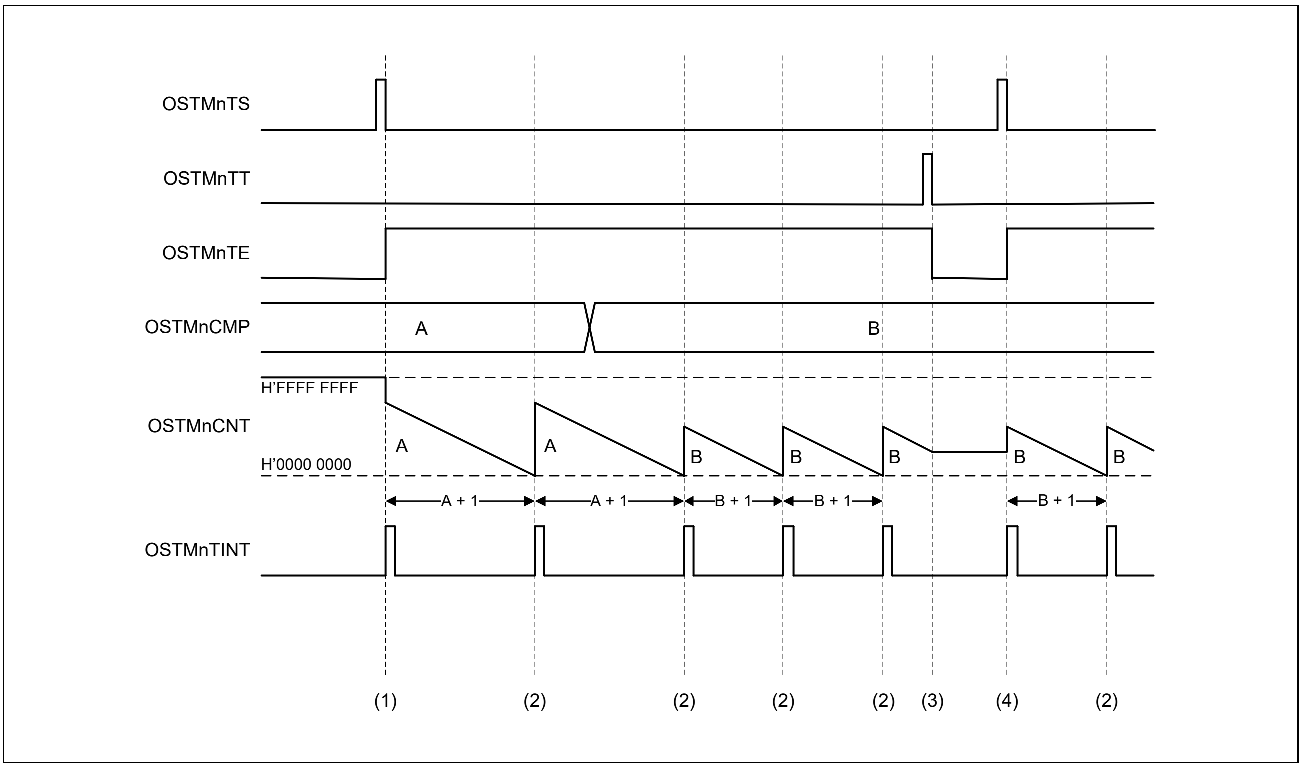


Figure 20.3 Timing Diagram of OSTM in Interval Timer Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter.
The counter starts counting down from the value of OSTMnCMP.
If OSTMnCTL.OSTMnMD0 is 1, OSTMTINT interrupt requests are generated at the start of counting. The OSTMnCNT register contains the current value as the counter.

(2) When the counter reaches H’0000 0000, an OSTMTINT interrupt request is generated. The counter loads the new start value from OSTMnCMP and continues counting down.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter. The counter retains its current value until it is restarted.

(4) When counting is restarted (OSTMnTS.OSTMnTS = 1), the counter loads the new start value from OSTMnCMP and starts counting down.

Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 during counting.

The counter loads the start value from the OSTMnCMP register and continues to count down.

The following figure shows the forced restart of the General Timer in interval timer mode, with counter-start interrupts enabled (OSTMnCTL.OSTMnMD0 = 1).

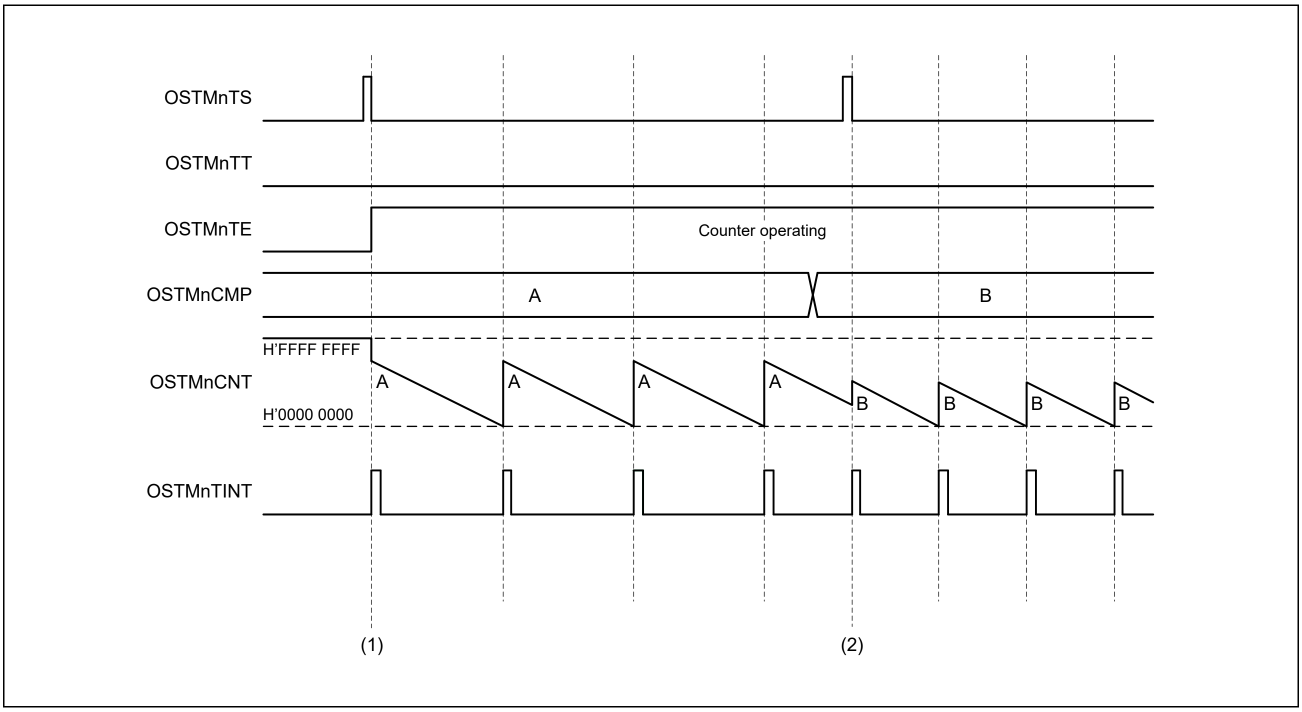


Figure 20.4 Timing Diagram of Forced Restart in Interval Timer Mode

Operations shown in the above timing diagram are as follows.

(1) The counter is started and stopped as described under **Figure 20.3, Timing Diagram of OSTM in Interval Timer Mode**.

(2) Setting OSTMnTS.OSTMnTS = 1 restarts the counter while counting is in progress (i.e. while OSTMnTE.OSTMnTE = 1).
The counter immediately restarts counting down, starting with the current value of OSTMnCMP.
When OSTMnCTL.OSTMnMD0 = 1, an OSTMTINT interrupt request is generated when counting starts.

20.3.5.2 Operation when OSTMnCMP = H’0000 0000

When OSTMnCMP = H’0000 0000, OSTM behaves as follows.

- When the counter is enabled, the OSTMTINT interrupt request is always set to 1.

The following figure shows operations of OSTM when OSTMnCMP = H’0000 0000, and counter-start interrupts are enabled.

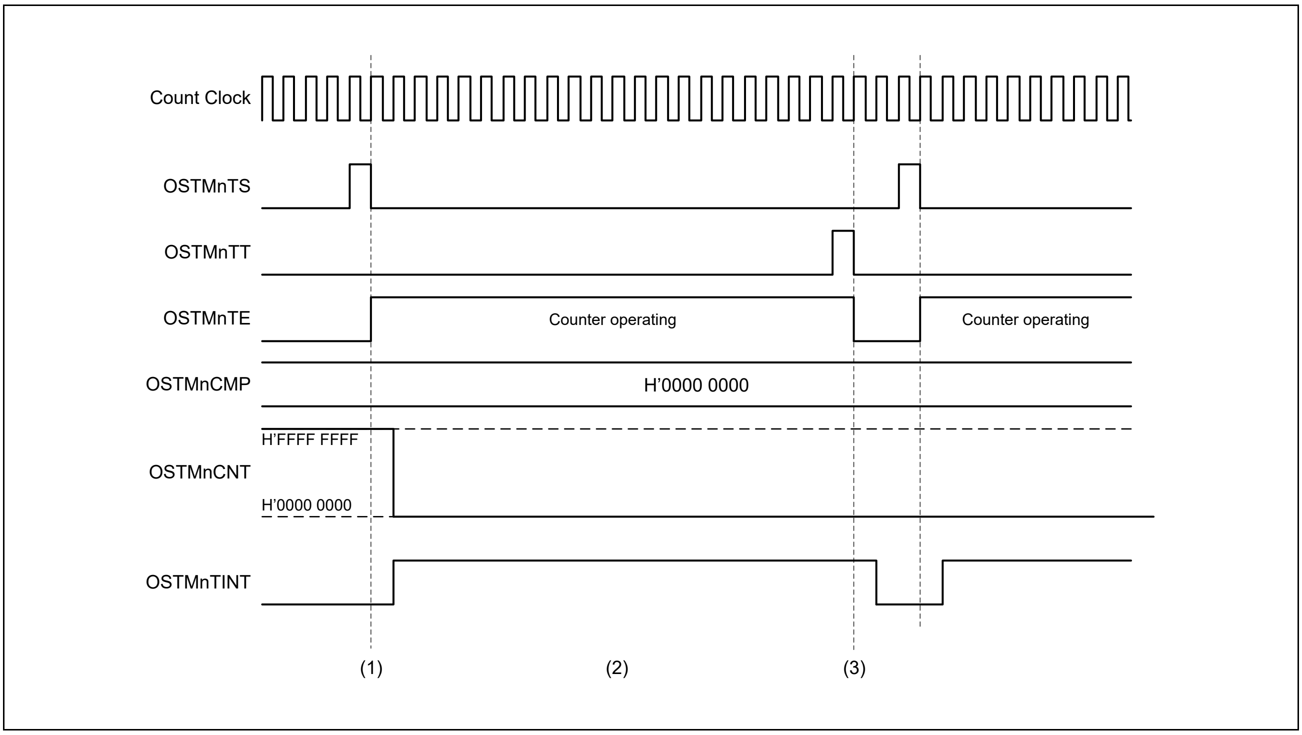


Figure 20.5 Timing Diagram when OSTMnCMP = H’0000 0000 in Interval Timer Mode

The timing diagram above shows the following operations:

(1) The counter is reloaded with the value in OSTMnCMP as soon as it starts counting, so the value H’0000 0000 is retained in OSTMnCMP.

(2) The OSTMTINT interrupt request is continuously asserted.

(3) After the counter stops, the OSTMTINT interrupt request signal is deasserted.

(4) When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

20.3.6 Free-Running Comparison Mode

20.3.6.1 Basic Operation in Free-Running Comparison Mode

In free-running comparison mode, the counter counts up from H’0000 0000 to H’FFFF FFFF. An OSTMnTINT interrupt request is output when the current value of the counter matches the value of the OSTMnCMP register. The free-running comparison mode is selected by setting the OSTMnCTL.OSTMnMD1 bit to 1.

New values can be written to the OSTMnCMP register at any time.

The following figure shows the basic operation of OSTM in free-run compare mode with the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

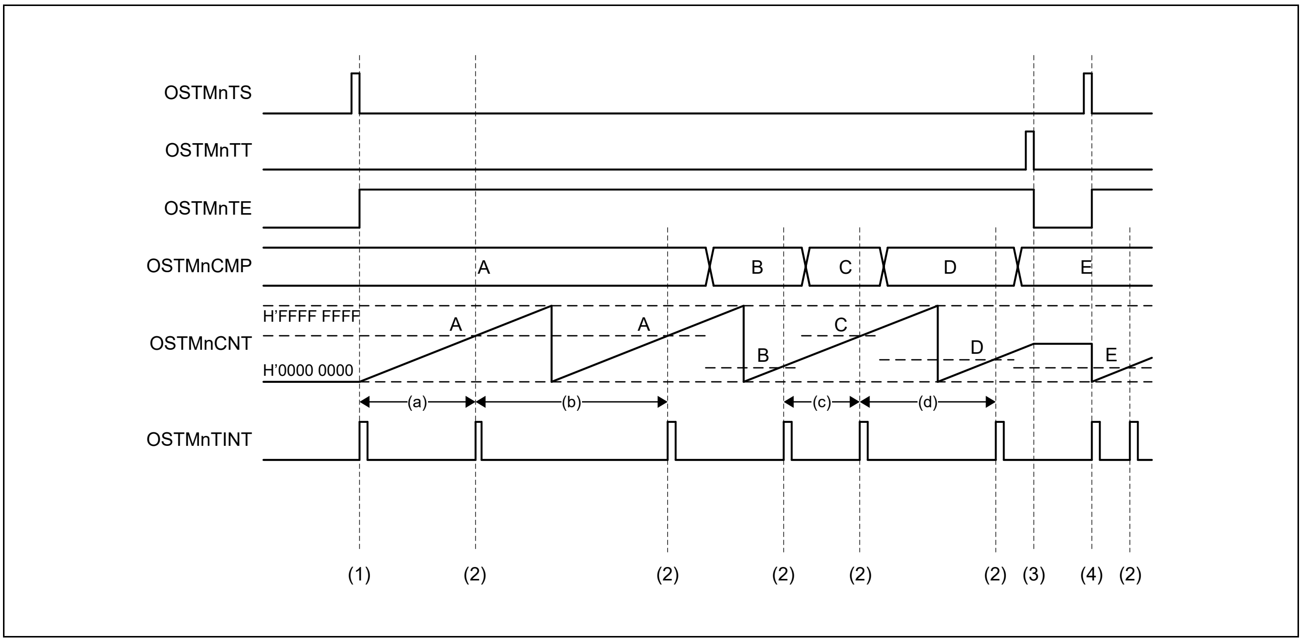


Figure 20.6 Timing Diagram of OSTM in Free-Run Compare Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1.
The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter. The counter counts up from H’0000 0000 to H’FFFF FFFF. The OSTMnCNT register is the counter, so it contains the current value.
When OSTMnCTL.OSTMnMD0 = 1, an OSTMTINT interrupt request is generated at the start of counting.

(2) When the current counter value matches the value in the OSTMnCMP register, an OSTMTINT interrupt request is generated.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter.
The counter retains its current value until it is restarted.

(4) Counting by the counter restarts from H’0000 0000 when OSTMnTS.OSTMnTS = 1.

OSTMTINT period

The OSTMTINT generation period is different at the start of counting and depends on the old and new compare values if OSTMnCMP is rewritten during operation.

Table 20.5 OSTMTINT Generation Timing

Old Value for Comparison	New Value for Comparison	Counter Value at Time of Rewriting	Period of OSTMTINT Generation	Label in Timing Diagram
Counter starts			(A + 1) × counter clock period	(a)
A	A	No rewriting	(H'FFFF FFFF + 1) × counter clock period	(b)
B	C > B	B < counter value < C	(C – B) × counter clock period	(c)
C	D < C	Counter value > D,C	(H'FFFF FFFF – C + D + 1) × counter clock period	(d)

Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set. The counter ignores the attempted setting and continues counting.

20.3.6.2 Operation when OSTMnCMP = H’0000 0000

The following figure shows the operation of OSTM when OSTMnCMP = H’0000 0000, and counter-start interrupts are enabled (OSTMnCTL.OSTMnMD0 = 1).

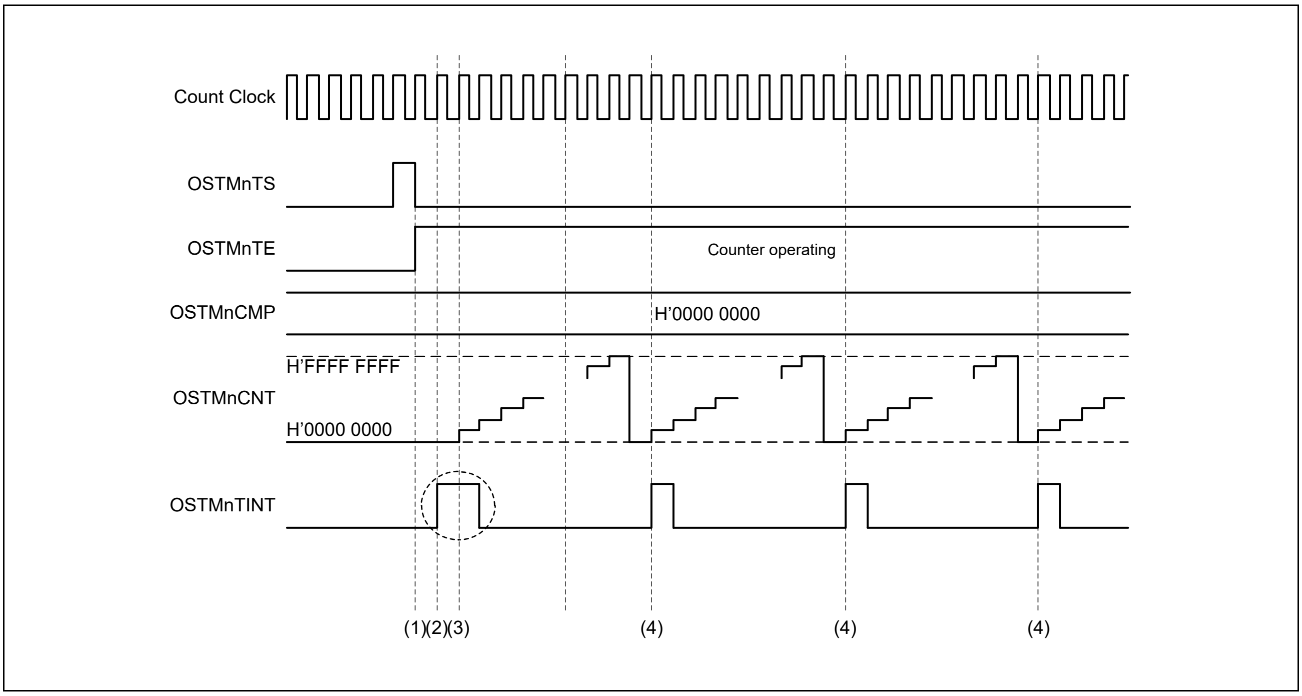


Figure 20.7 Timing Diagram when OSTMnCMP = H’0000 0000 in Free-Run Compare Mode

The timing diagram above shows the following operations.

(1) Once the counter starts, it counts up from H’0000 0000 to H’FFFF FFFF.

(2) An OSTMTINT interrupt request is generated when counting starts.

(3) If the current counter value matches OSTMnCMP, an OSTMTINT interrupt request is generated. If OSTMnCMP = H’0000 0000 in the above case, OSTMTINT is generated over two clock cycles.

(4) Every (H’FFFF FFFF + 1) clock cycles the OSTMTINT interrupt request is asserted.

When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

21. Watchdog Timer (WDT)

This LSI has 3 channels of watchdog timer as **Table 21.1** and generates a reset request signal when the counter value is not rewritten and overflows due to system runaway.

Table 21.1 WDT Channels

Channel	Functions
WDT CH0	WDT to check the operation of Cortex-A55-CPU Core0
WDT CH1	WDT to check the operation of Cortex-A55-CPU Core1
WDT CH2	WDT to check the operation of Cortex-M33 CPU

Also, a reset request signal or interrupt signal can be generated by inputting a parity error signal due to a parity error in the CPU of this LSI.

21.1 Features

All WDT modules have the following functions implemented in common as basic WDT functions.

21.1.1 Normal Watchdog Timer Function

- Bus interface: Compatible with AMBA® 2.0 APB.
- Operating frequency: Bus interface 100 [MHz], counter control 24 [MHz]
- The counter operation and the bus interface are asynchronous and can operate without depending on the size relationship of the clock cycles of each other.
- A 32-bit counter that operates with the clock (WDTn_CLK (n = 0, 1, 2)) input to this module.
- An interrupt request signal is generated every cycle set in the counter of this module.
- When WDTn_CLK (n = 0, 1, 2) = 24 MHz, the counter cycle can be set in 43.69 msec unit from 43.69 msec to 178956.97 msec.
- After the interrupt request signal is generated, if the software does not clear the watchdog timer until the next counter overflow, the reset request signal is generated.
- The set value of each register and the Elapsed time of the watchdog timer can be read.

21.1.2 Reset Request Function due to CPU Parity Error

- A reset request signal (WDTRSTB) or interrupt signal (PERROUT) is generated by inputting a parity error signal.
- The reset request signal is generated as the OR of the reset request of this module and the reset request of the parity error circuit.
- The polarity of the parity error signal can be set with this module.
- Function can be turned ON/OFF for each 32-bit parity error signal.
- The output of the reset request signal and the interrupt signal can be switched by inputting the parity error signal.
- It is possible to force a parity error to occur by register setting.

21.1.3 Internal Block Diagram

Figure 21.1 shows the internal block of WDT. WDT consists of a WDT-CORE part that implements the function of a normal watchdog timer and a part that detects a parity error.

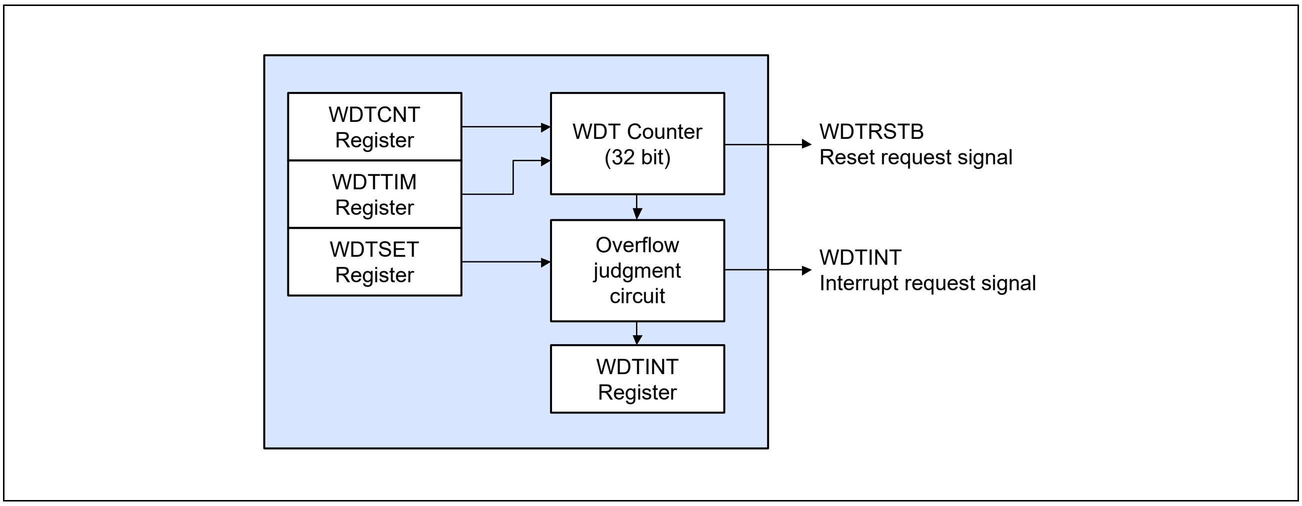


Figure 21.1 WDT_CORE Internal Block

In addition to the above WDT-Core function, WDT receives a parity error interrupt signal generated by the CPU when a Parity Error occurs in the CPU monitored by WDT and generates a reset request signal or interrupt request signal.

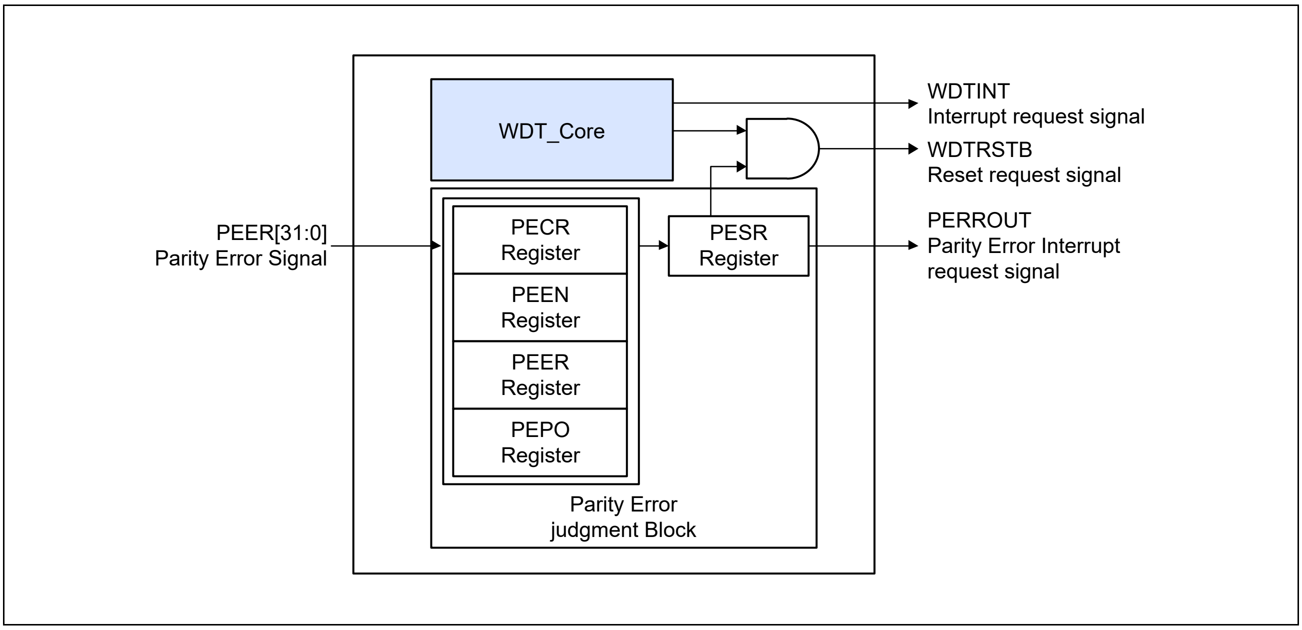


Figure 21.2 WDT_CORE and Parity Error handling block

21.1.4 Input/Output Pin

Table 21.2 shows the pin configuration.

Table 21.2 Pin Configuration

Watchdog timer overflow/ CPU parity error	WDTOVF_PERROUT#	Output	Watchdog timer counter overflow signal output/ Interrupt signal output when CPU parity error occurs

21.2 Register Configuration

Table 21.3 shows the base address for each WDT channel.

Table 21.3 Base Address for each WDT channel

	Channel	WDT	Base Address
Cortex-A55 Address Space	WDT CH0	WDT to check the operation of Cortex-A55-CPU Core0	H'0_1280_0800
	WDT CH1	WDT to check the operation of Cortex-A55-CPU Core1	H'0_1280_0C00
	WDT CH2	WDT to check the operation of Cortex-M33 CPU	H'0_1280_0400
Cortex-M33 Address Space Non-Secure	WDT CH0	WDT to check the operation of Cortex-A55-CPU Core0	H'4280_0800
	WDT CH1	WDT to check the operation of Cortex-A55-CPU Core1	H'4280_0C00
	WDT CH2	WDT to check the operation of Cortex-M33 CPU	H'4280_0400
Cortex-M33 Address Space Secure	WDT CH0	WDT to check the operation of Cortex-A55-CPU Core0	H'5280_0800
	WDT CH1	WDT to check the operation of Cortex-A55-CPU Core1	H'5280_0C00
	WDT CH2	WDT to check the operation of Cortex-M33 CPU	H'5280_0400

Table 21.4 shows WDT Register Configuration on all WDT Channels.

Table 21.4 WDT Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
WDT CH0	WDT Control Register_0	WDTCNT_0	R/W	H’0000_0000	H’0000	32
	WDT Period Setting Register_0	WDTSET_0	R/W	H’FFF0_0000	H’0004	32
	WDT Elapsed Time Register_0	WDTTIM_0	R/W	H’0000_0000	H’0008	32
	WDT Interrupt control Register_0	WDTINT_0	R/W	H’0000_0000	H’000C	32
	Parity Error Control Register_0	PECR_0	R/W	H’0000_0000	H’0010	32
	Parity Error forced Enable Register_0	PEEN_0	R/W	H’0000_0000	H’0014	32
	Party Status Register_0	PESR_0	R/W	H’0000_0000	H’0018	32
	Parity Error Enable Register_0	PEER_0	R/W	H’0000_0000	H’001C	32
	Parity Error Polarity setting Register_0	PEPO_0	R/W	H’0000_0000	H’0020	32
WDT CH1	WDT Control Register_1	WDTCNT_1	R/W	H’0000_0000	H’0000	32
	WDT Period Setting Register_1	WDTSET_1	R/W	H’FFF0_0000	H’0004	32
	WDT Elapsed Time Register_1	WDTTIM_1	R/W	H’0000_0000	H’0008	32
	WDT Interrupt control Register_1	WDTINT_1	R/W	H’0000_0000	H’000C	32
	Parity Error Control Register_1	PECR_1	R/W	H’0000_0000	H’0010	32
	Parity Error forced Enable Register_1	PEEN_1	R/W	H’0000_0000	H’0014	32
	Party Status Register_1	PESR_1	R/W	H’0000_0000	H’0018	32
	Parity Error Enable Register_1	PEER_1	R/W	H’0000_0000	H’001C	32
	Parity Error Polarity setting Register_1	PEPO_1	R/W	H’0000_0000	H’0020	32
WDT CH2	WDT Control Register_2	WDTCNT_2	R/W	H’0000_0000	H’0000	32
	WDT Period Setting Register_2	WDTSET_2	R/W	H’FFF0_0000	H’0004	32
	WDT Elapsed Time Register_2	WDTTIM_2	R/W	H’0000_0000	H’0008	32
	WDT Interrupt control Register_2	WDTINT_2	R/W	H’0000_0000	H’000C	32
	Parity Error Control Register_2	PECR_2	R/W	H’0000_0000	H’0010	32
	Parity Error forced Enable Register_2	PEEN_2	R/W	H’0000_0000	H’0014	32
	Party Status Register_2	PESR_2	R/W	H’0000_0000	H’0018	32
	Parity Error Enable Register_2	PEER_2	R/W	H’0000_0000	H’001C	32
	Parity Error Polarity setting Register_2	PEPO_2	R/W	H’0000_0000	H’0020	32

21.3 Register Descriptions

21.3.1 WDT Control Register_n (WDTCNT_n) (n = 0, 1, 2)

This register sets the operation enable of the watchdog timer function. Once the software activates the watchdog timer, the watchdog timer does not stop*[1] until it is reset. Also, the setting register*[2] of the watchdog timer cannot be changed. The setting register change must be completed*[2] before enabling the watchdog timer operation.

Note 1. For debugging, the internal counter can be paused with the CNTSTOP signal.

Note 2. WDT cycle setting register (WDTSET) and WDT Elapsed time register (WDTTIM)

Note 3. When changing the setting register, wait until the write data value of the setting register is reflected before enabling the watchdog timer operation. The reflection time of the setting register is 6 × P0ϕ + 9 × OSCCLK or more. If this reflection time is not observed, the setting register value will not be reflected and will be ignored.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDTEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	WDTEN	0	R/W	Watchdog timer enable register 1: Watchdog timer operation enabled 0: Disabled

21.3.2 WDT Period Setting Register_n (WDTSET_n) (n = 0, 1, 2)

This register is used to set the cycle of the 32-bit counter that composes the watchdog timer. Writing is not possible while the watchdog timer is running. If you write continuously, the data written later will be valid. Calculate the watchdog timer cycle using the following formula.

$$\text{Watchdog timer cycle} = \text{WDTn\_CLK } (\text{n} = 0{,}1{,}2) \text{ cycle} \times 1024 \times 1024 \times (\text{WDTTIME setting value} + 1)$$

$$\text{WDTTIME setting value} = \frac{\text{WDT cycle}}{(\text{WDTn\_CLK } (\text{n} = 0{,}1{,}2) \text{ cycle} \times 1024 \times 1024) - 1}$$

For example, WDTn_CLK (n = 0, 1, 2) = 24 MHz, the counter cycle can be set in 43.69 msec unit from 43.69 msec to 178956.97 msec.

It can be set in units. The setting value of WDTTIME is the value set in WDTSET[31:20], and the value from H’000 to H’FFF can be set.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDTTIME11	WDTTIME10	WDTTIME9	WDTTIME8	WDTTIME7	WDTTIME6	WDTTIME5	WDTTIME4	WDTTIME3	WDTTIME2	WDTTIME1	WDTTIME0	—	—	—	—
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	WDTTIME [11:0]	All 1	R/W	Watchdog timer period cycle setting It cannot be written while the watchdog timer is running.
19 to 0	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.

CAUTION

The value that the watchdog timer compares when an interrupt occurs is “WDTSET[31:0] + H’000F_FFFF”. Therefore, when the setting value of this register is H’0010_0000, the interrupt occurrence count is “H’001F_FFFF” (not H’0010_0000).

21.3.3 WDT Elapsed Time Register_n (WDTTIM_n (n = 0, 1, 2))

This register indicates the 32-bit count value that composes the watchdog timer. If the watchdog timer is read during operation, the Elapsed time from the time it is cleared to the time of reading can be read.

By writing to this register when the watchdog timer is stopped, the value at the start of counting can be set in the 32-bit counter that configures the watchdog timer. When a value larger than the counter cycle (value when clearing) set in the WDT cycle setting register is written and the watchdog timer operation is started, the 32-bit counter once overflows and returns to 0. It operates at the set correct cycle.

You cannot write while the watchdog timer is running. If you write continuously, the data written later will be valid.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTTIME31	CRTTIME30	CRTTIME29	CRTTIME28	CRTTIME27	CRTTIME26	CRTTIME25	CRTTIME24	CRTTIME23	CRTTIME22	CRTTIME21	CRTTIME20	CRTTIME19	CRTTIME18	CRTTIME17	CRTTIME16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTTIME15	CRTTIME14	CRTTIME13	CRTTIME12	CRTTIME11	CRTTIME10	CRTTIME9	CRTTIME8	CRTTIME7	CRTTIME6	CRTTIME5	CRTTIME4	CRTTIME3	CRTTIME2	CRTTIME1	CRTTIME0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTTIME [31:0]	All 0	R/W	Watchdog timer count value. It cannot be written while the watchdog timer is running.

CAUTION

When setting a value in the WDTTIM register, do not set a value that is WDTSET[31:0] register + H’000F_FFFF.

21.3.4 WDT Interrupt Control Register_n (WDTINT_n (n = 0, 1, 2))

This register reads the interrupt status of the watchdog timer, clears interrupts, and clears the counters that compose the watchdog timer. Set the INTDISP bit of this register to “1” within the time set by WDTSET. If not set, the WDTINT pin is asserted. After that, within the time set by WDTSET

When the INTDISP bit is not set to “1”, the WDTRSTB pin is asserted.

When performing continuous write access to the WDTINT register, leave a write interval of 5 × P0ϕ + 5 × OSCCLK or more. If the write interval is not observed, the register will not be written.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTDISP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	INTDISP	0	R/W	Register for interrupt status read and clear Read: Interrupt status 1: With interrupt 0: Without interrupt Write: Interrupt clear 1: Interrupt clear 0: Invalid (no change)

21.3.5 Parity Error Control Register_n (PECR_n (n = 0, 1, 2))

This register controls whether a reset (WDTRSTB) or an interrupt (PERROUT) is generated due to a CPU parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PECR31	PECR30	PECR29	PECR28	PECR27	PECR26	PECR25	PECR24	PECR23	PECR22	PECR21	PECR20	PECR19	PECR18	PECR17	PECR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PECR15	PECR14	PECR13	PECR12	PECR11	PECR10	PECR9	PECR8	PECR7	PECR6	PECR5	PECR4	PECR3	PECR2	PECR1	PECR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PECR [31:0]	All 0	R/W	Parity error control Controls the following operations when the value is 1 for each bit of PESR[31:0] 0: Reset (WDTRSTB) is generated 1: Generate interrupt (PEEROUT)

21.3.6 Parity Error Forced Enable Register_n (PEEN_n (n = 0, 1, 2))

This register forcibly asserts reset (WDTRSTB) or PERROUT even when no CPU parity error has occurred.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PEEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	PEEN	0	R/W	Parity error forced enable 0: Reset (WDTRSTB) or PERROUT generation follows the operation of the parity error register 1: Force reset (WDTRSTB) or assert PERROUT

21.3.7 Parity Error Status Register_n (PESR_n (n = 0, 1, 2))

This register holds the cause of CPU parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PESR31	PESR30	PESR29	PESR28	PESR27	PESR26	PESR25	PESR24	PESR23	PESR22	PESR21	PESR20	PESR19	PESR18	PESR17	PESR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PESR15	PESR14	PESR13	PESR12	PESR11	PESR10	PESR9	PESR8	PESR7	PESR6	PESR5	PESR4	PESR3	PESR2	PESR1	PESR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PESR[31:0]	All 0	R/W	Parity error flag 0 to 31 Indicates that a parity error has occurred. Read: Interrupt status 1: With interrupt 0: Without interrupt Write: Interrupt clear 1: Interrupt clear 0: Invalid (nothing changed)

21.3.8 Parity Error Enable Register_n (PEER_n (n = 0, 1, 2))

This register controls the assertion of the WDTRSTB signal and PERROUT signal due to a CPU parity error. Each bit controls the behavior of parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PEER31	PEER30	PEER29	PEER28	PEER27	PEER26	PEER25	PEER24	PEER23	PEER22	PEER21	PEER20	PEER19	PEER18	PEER17	PEER16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEER15	PEER14	PEER13	PEER12	PEER11	PEER10	PEER9	PEER8	PEER7	PEER6	PEER5	PEER4	PEER3	PEER2	PEER1	PEER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PEER[31:0]	All 0	R/W	Parity error enable 31-0 Controls the operation of PESR. 0: Disables reset (WDTRSTB) and PERROUT generation operations 1: Permits reset (WDTRSTB) and PERROUT generation operation

21.3.9 Parity Error Polarity Setting Register_n (PEPO_n (n = 0, 1, 2))

This register sets the polarity of CPU parity error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PEPO31	PEPO30	PEPO29	PEPO28	PEPO27	PEPO26	PEPO25	PEPO24	PEPO23	PEPO22	PEPO21	PEPO20	PEPO19	PEPO18	PEPO17	PEPO16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEPO15	PEPO14	PEPO13	PEPO12	PEPO11	PEPO10	PEPO9	PEPO8	PEPO7	PEPO6	PEPO5	PEPO4	PEPO3	PEPO2	PEPO1	PEPO0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PEPO [31:0]	All 0	R/W	Active polarity of parity 31-0 Set the Active polarity (H/L) of each parity error signal connected to PEERR. 0: Active High 1: Active Low

21.3.10 Register Setting Order for Each WDT Channel

The following shows the register setting sequence for controlling the watchdog timer and parity error circuit of each channel. The control of the original watchdog timer and the parity error circuit do not affect each other, so there is no restriction on the setting order in the following diagrams (1) and (2).

The parity error circuit cannot accept the parity error that occurred before setting PEER = 1. When rewriting PEEN from 0 to 1, it is possible to rewrite at any timing. However, when rewriting PEEN from 1 to 0, once PEER = 0 and (2) flow execution is required.

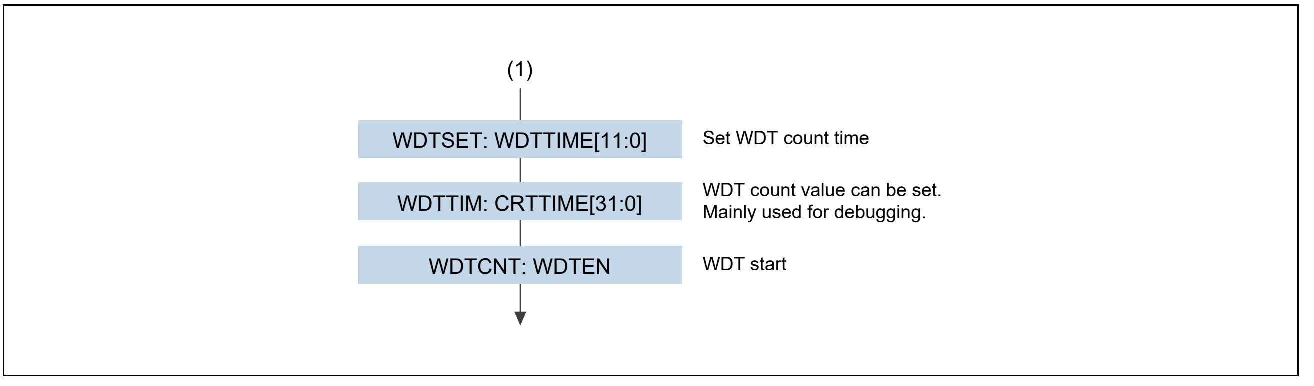


Figure 21.3 WDT Register Setting Order

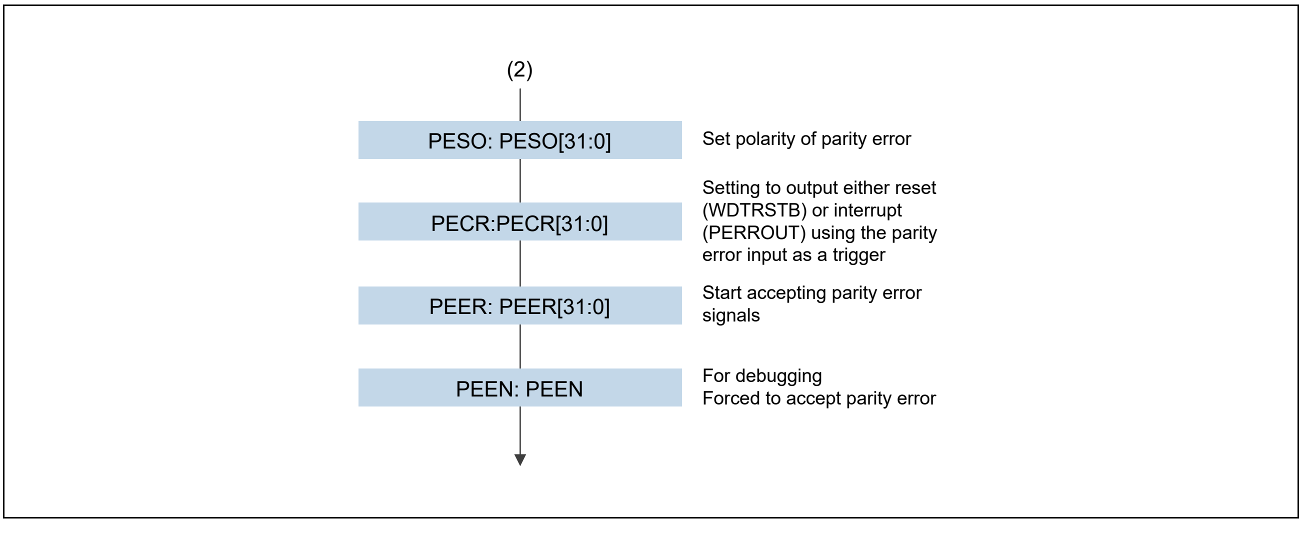


Figure 21.4 Parity Error Circuit Register Setting Order

21.4 Operation

21.4.1 WDT_CORE Operaton Timing

Figure 21.5 shows the operation diagram of the WDT-Core block. When 1 is set in Bit [0] of the WDTCNT register, the 32-bit internal counter WDT-Counter starts counting. If the counter value matches the value set in the WDTSET register, the counter becomes overflow, clears the counter to 0, and then generates an interrupt request signal.

An interrupt signal is output at the first counter overflow, and a reset request signal is output if the counter is not cleared and the second counter overflow occurs.

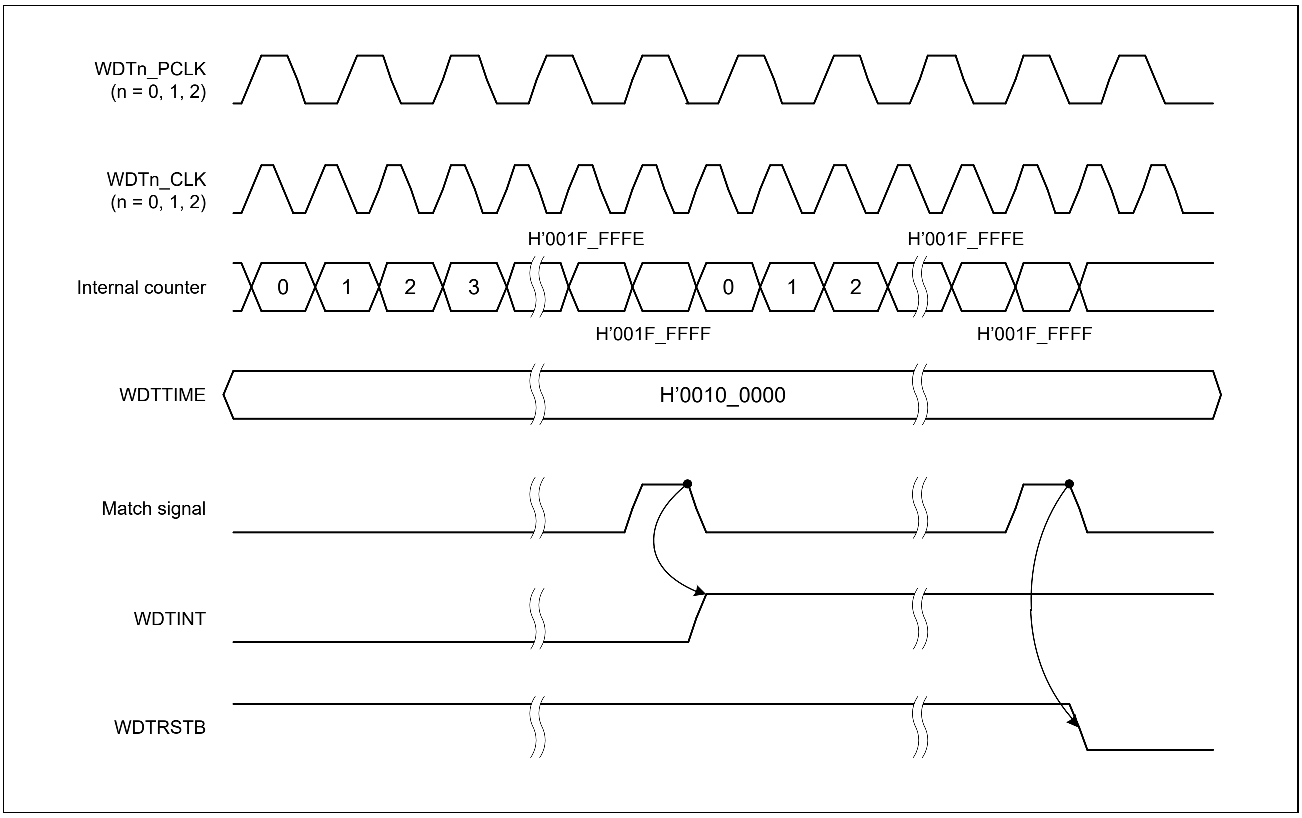


Figure 21.5 WDT Output Signal Operation Timing

21.4.2 WDT Interrupt Control Register (WDTINT) Specification for Write Access Interval

After the write access to the WDT interrupt control register is completed, the written data is synchronized with WDTn_CLK (n = 0, 1, 2), so a synchronization period is required. When performing write access to the WDTINT register, it is necessary to leave a write interval of 5 × P0ϕ + 5 × OSCCLK or more between the next write access.*1

Note 1. If the write interval is not observed, the interrupt request may not be cleared. In this case, after making a write access and confirming that the interrupt factor has been cleared, perform the following write access.

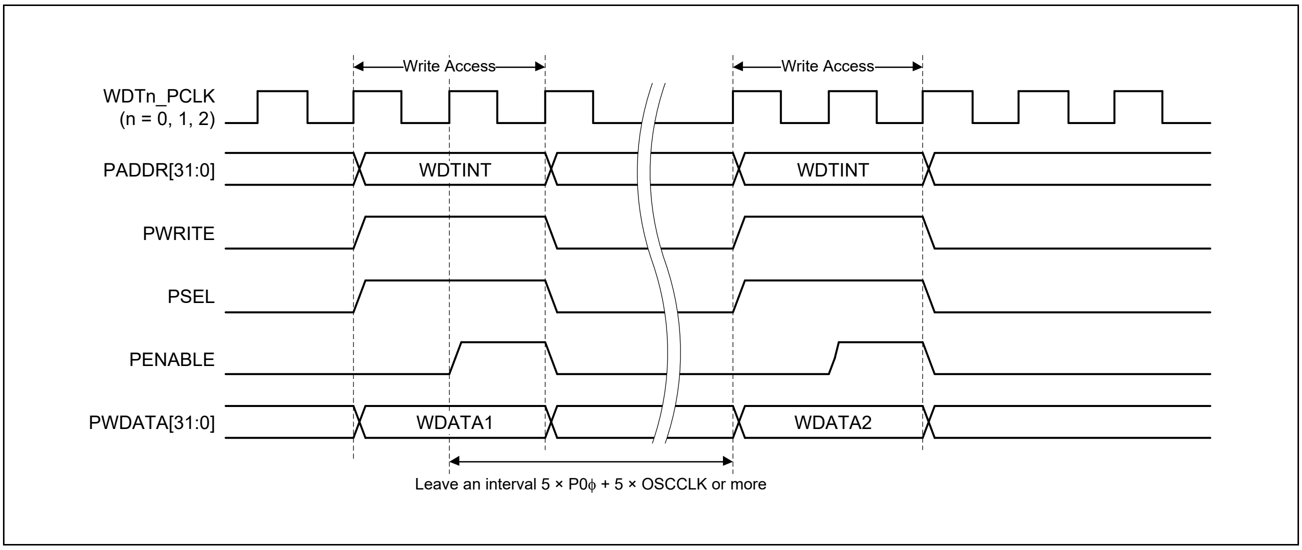


Figure 21.6 WDTINT Write Access Interval Regulation

21.4.3 Watchdog Timer Operation Reflection Timing of Register Setting Value

Each register other than the WDT interrupt control register (WDTINT) is always synchronized with WDTn_CLK (n = 0, 1, 2) at regular intervals. The timing at which the written data is reflected in the watchdog timer operation is in the range of small 2 × P0ϕ + 4 × OSCCLK) to large value (6 × P0ϕ +9 × OSCCLK).

Figure 21.7 (Watchdog timer operation reflection timing 1 of register setting value) shows how write data is small and synchronized.

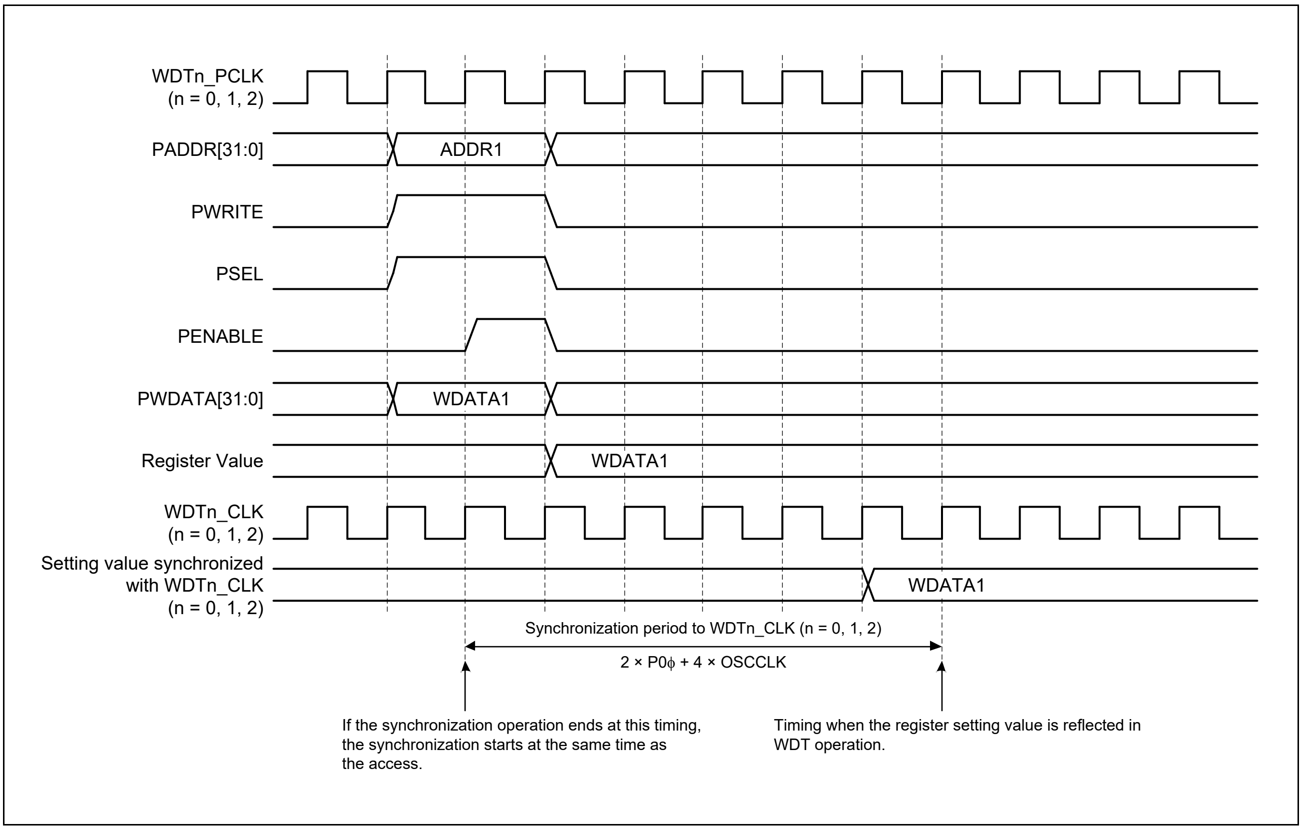


Figure 21.7 Watchdog Timer Operation Reflection Timing of Register Setting Value - 1

In addition, **Figure 21.8** (Watchdog timer operation reflection timing 2 of register setting value) shows how write data is large and synchronized.

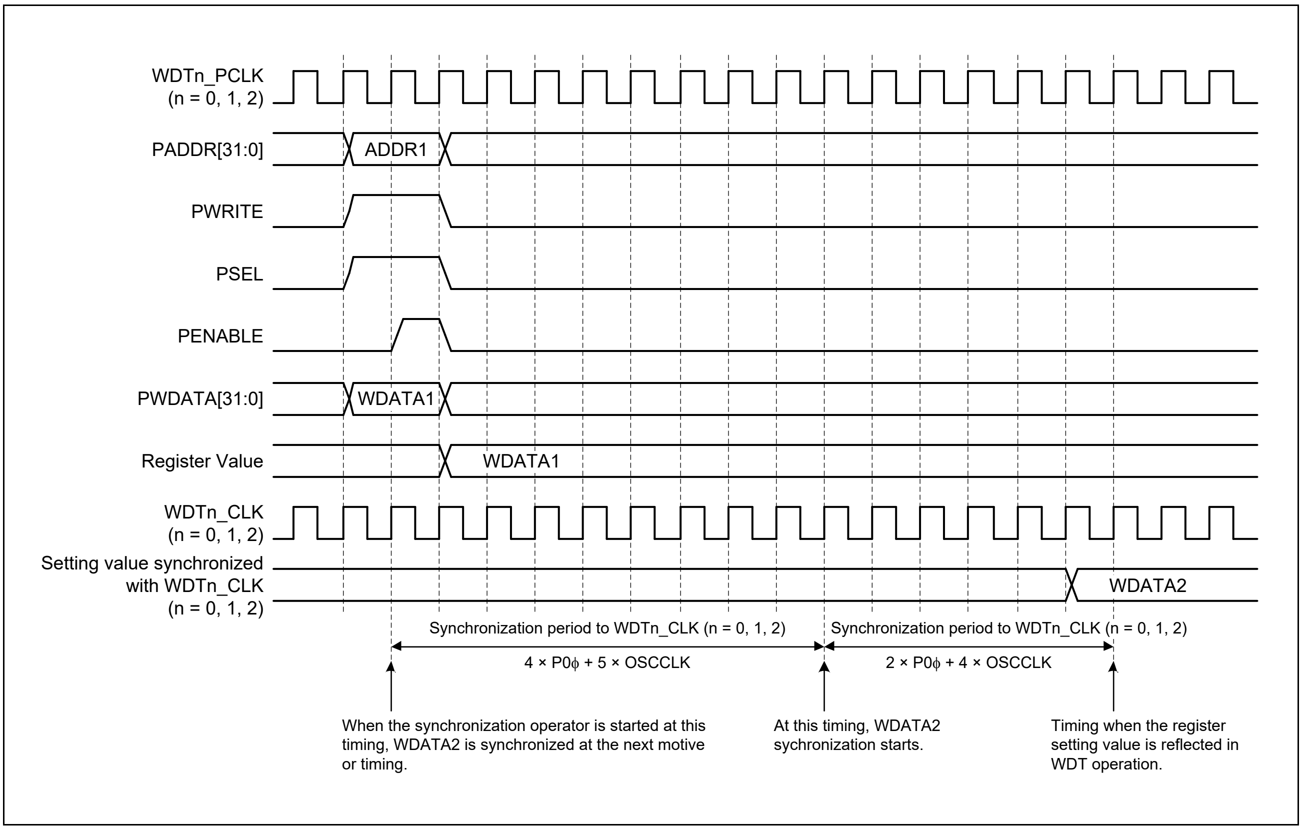


Figure 21.8 Watchdog Timer Operation Reflection Timing of Register Setting Value - 2

21.4.4 Timing of Reflecting WDT Counter Value to WDTTIM Register

The value stored in the WDTTIM register is the value obtained by synchronizing the counter value with P0φ. Therefore, the interval at which the value is updated in the WDTTIM register is 5 × OSCCLK+5 × P0ϕ at most, and the value before the actual counter value is read.

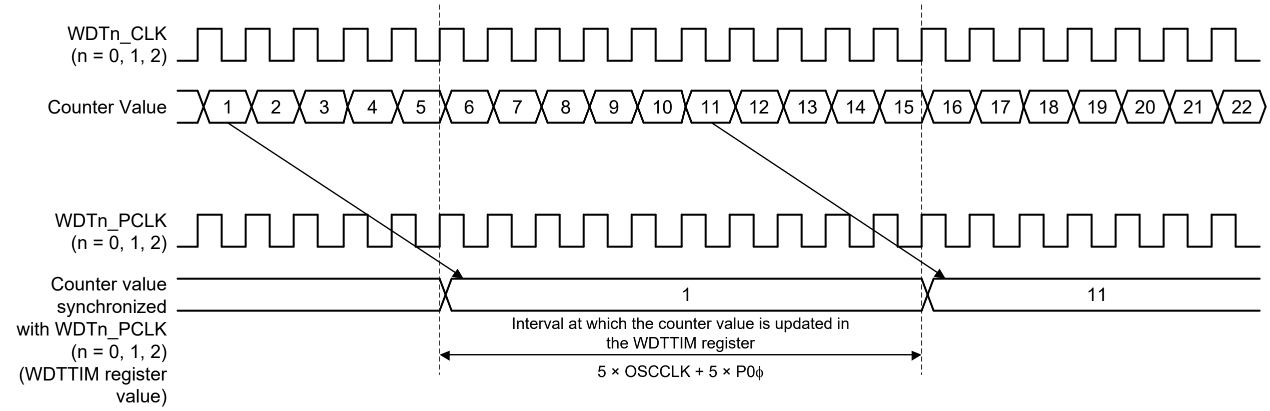


Figure 21.9 Timing of Reflecting the Counter Value to the WDTTIM Register

21.4.5 Conditions Under which WDTRSTB and PERROUT Operate

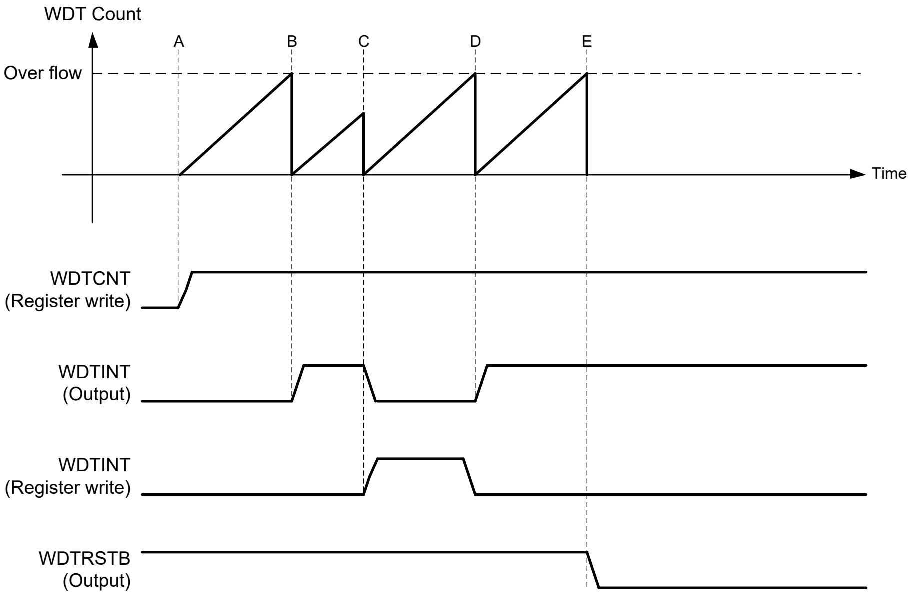
Table 21.5 shows the conditions under which WDTRSTB and PERROUT operate.

When PEER = 0, WDTRSTB follows the watchdog timer operation described in **Section 21.4.1** to **Section 21.4.5** regardless of the occurrence of parity error. PERROUT holds 0. (No. 1, No. 2) No. 3 to No. 6 show the operating conditions when the parity error input is enabled by PEER = 1. When a parity error occurs (No. 3, No. 4), PESR holds the error status, and WDTRSTB or PERROUT is asserted depending on the value of PECR. If no parity error occurs, PESR = 0, WDTRSTB propagates the WDT-Core logic, and PERROUT is in the unasserted state. (No. 5, No. 6) When PEEN = 1 forces a parity error to occur, WDTRSTB or PERROUT is asserted depending on the PECR value regardless of the parity error.

Table 21.5 WDTRSTB/PERROUT Operating Conditions

No.	PEERR	PEPO	PEER	PEEN	PECR	PESR	WDTRSTB	PERROUT
1	X	1/0	0	0	0	0 (Fixed)	Original WDT	0: de-assert
2		1/0	0	0	1	0 (Fixed)	Original WDT	0: de-assert
3	With parity	1/0	1	0	0	1	0: assert	0: de-assert
4		1/0	1	0	1	1	Original WDT	1: assert
5	No parity	1/0	1	0	0	0	Original WDT	0: de-assert
6		1/0	1	0	1	0	Original WDT	0: de-assert
7	X	X	X	1	0	X	0: assert	0: de-assert
8		X	X	1	1	X	Original WDT	1: assert

The timing waveforms for No. 1 to No. 4 in **Table 21.5** are shown below.



Note: A) Set WDTCNT = 1 to start WDT counting.

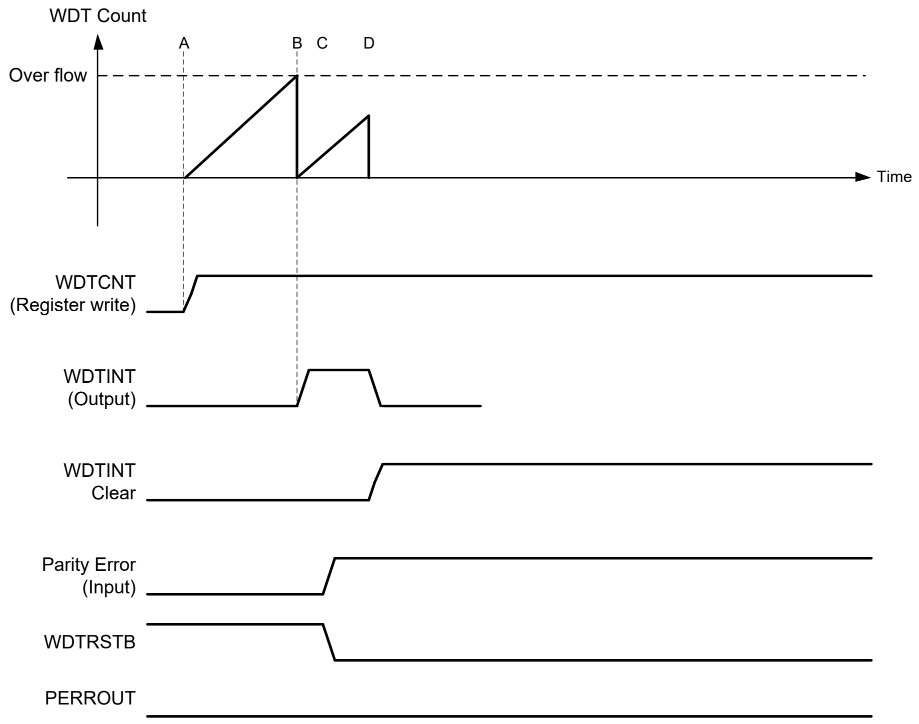
B) The interrupt request signal (WDTINT) is asserted at the first overflow of the WDT count.

C) If WDTINT (INTDISP) = 1 is written before the WDT count overflows, the interrupt request is cleared and the WDT count becomes zero.

D) Again, the interrupt request signal is asserted at the first overflow.

E) When the WDT count overflows for the second time, WDTRSTB is asserted.

Figure 21.10 Normal Watchdog Timer Operation No. 1/No. 2



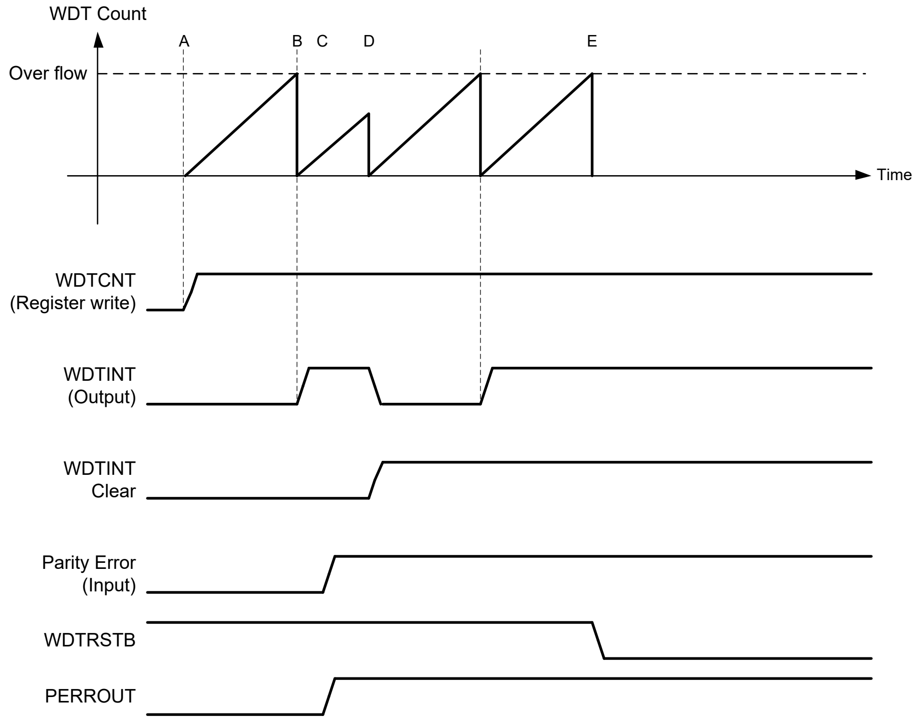
Note: A) Set WDTCNT = 1 to start WDT counting.

B) The interrupt request signal (WDTINT) is asserted at the first overflow of the WDT count.

C) When a parity error occurs, WDTRSTB = 0. At the same time, hold PERROUT = 0.

D) Even if WDT Count is reset by WDTINT = 1, WDTRSTB = 0 and PERROUT = 0 are retained.

Figure 21.11 Watchdog Timer Operation when a Parity Error Signal Occurs No. 3



Note: A) Set WDTCNT = 1 to start WDT counting.

B) The interrupt request signal (WDTINT) is asserted at the first overflow of the WDT count.

C) When a parity error occurs, WDTRSTB = 1. At the same time, hold PERROUT = 1.

D) Even if WDT Count is reset by WDTINT = 1, WDTRSTB = 1 and PERROUT = 1 are retained.

E) WDTRSTB = 0 is asserted due to WDT Count overflow.

Figure 21.12 Watchdog Timer Operation when a Parity Error Signal Occurs No. 4

21.5 Reset Control

21.5.1 WDTRST Control of Each WDT Channels

WDT CH0 generates WDTINT every cycle set in WDTSET. The CPU (Cortex-A55-Core0) receives the WDT CH0 interrupt signal WDTINT, clears the WDT CH0 counter at regular intervals, and suppresses the overflow of the WDT counter.

WDT CH1 generates WDTINT every cycle set in WDTSET. The CPU (Cortex-A55-Core1) receives the WDT CH1 interrupt signal WDTINT and clears the WDT CH1 counter at regular intervals to suppress the overflow of the WDT counter.

WDT CH2 generates WDTINT every cycle set in WDTSET. The CPU (Cortex-M33) receives the WDT CH2 interrupt signal WDTINT and clears the WDT CH2 counter at regular intervals to prevent the WDT counter from overflowing.

If the counter is not cleared before the cycle set for each WDT elapses, the uncleared WDT sends a reset request signal to the CPG. Based on the setting of the built-in register CPG_WDTRST_SEL, CPG can select whether to reset only the CPU corresponding to the WDT that caused the reset from the WDT reset request signal from each channel, or to reset the entire system. is. You can also choose whether to assert the WDTOVF_PERROUT# pin to notify the outside world that the entire system has been reset.

(For details about CPG_WDTRST_SEL register in CPG, refer to the CPG chapter.)

Table 21.6 WDT Reset Target by CPG_WDTRST_SEL Register

Bit	Initial Value	R/W	Reset Factor	Function
CPG_WDTRST_SEL[0]	0	R/W	WDT CH0	When the target WDT executes a reset request due to overflow etc. 0: Mask system reset 1: System reset implementation
CPG_WDTRST_SEL[1]	0	R/W	WDT CH1	
CPG_WDTRST_SEL[2]	0	R/W	WDT CH2	
CPG_WDTRST_SEL[4]	0	R/W	WDT CH0	When the target WDT executes a reset request due to overflow etc. 0: Mask WDTOVF_PERROUT assert 1: WDTOVF_PERROUT assert
CPG_WDTRST_SEL[5]	0	R/W	WDT CH1	
CPG_WDTRST_SEL[6]	0	R/W	WDT CH2	
CPG_WDTRST_SEL[8]	0	R/W	WDT CH0	When either WDT CH0 or WDT CH1 makes a reset request 0: Mask the Cold-Reset of Cortex-A55 1: Implemented Cortex-A55 Cold-Reset
CPG_WDTRST_SEL[9]	0	R/W	WDT CH1	
CPG_WDTRST_SEL[10]	0	R/W	WDT CH2	0: Mask the Cold reset of the paired Cortex-M33 1: Perform Cold reset

When the WDT system reset is executed, it is also possible to set the CPG_WDTRST_SEL[6:4] bit to notify the system outside the LSI from the WDTOVF_PERROUT# pin of the GPIO pin that the system reset has occurred due to the WDT overflow. When using WDTOVF_PERROUT# as a signal to notify the system reset to the outside, set the CPG_WDTRST_SEL[6:4] bit to 1 when CPG_WDTRST_SEL[2:0] is 1 as shown below.

CPG_WDTRST_SEL[4]	Set 1 when CPG_WDTRST_SEL[0] = 1
GPG_WDTRST_SEL[5]	Set 1 when CPG_WDTRST_SEL[1] = 1
CPG_WDTRST_SEL[6]	Set 1 when CPG_WDTRST_SEL[2] = 1

For example, if CPG_WDTRST_SEL[0] = 0 and WDTRST_SEL[4] = 1 are set, the system reset will not be executed even if WDT CH0 asserts a reset request because CPG_WDTRST_SEL[0] = 0, but WDTRST_SEL[4] = 1 so WDTOVF_PERROUT# pin will be asserted. In other words, the system reset execution status and WDTOVF_PERROUT do not match.

21.6 Usage Note

[Cautions]

- When setting a value in the WDTTIM register, do not set a value that is WDTSET register + H'F_FFFF.
- The reset request signal WDTRSTB and WDT interrupt signal WDTINT cannot be masked.

22. Serial Communications Interface with FIFO (SCIFA)

This LSI has five channels of serial communication interface (SCIFA) with FIFO that support both asynchronous and clock synchronous serial communication. The SCIFA has 16-stage FIFO buffers for transmission and reception, respectively, for each channel that enable this LSI to perform efficient high-speed continuous communication.

22.1 Overview

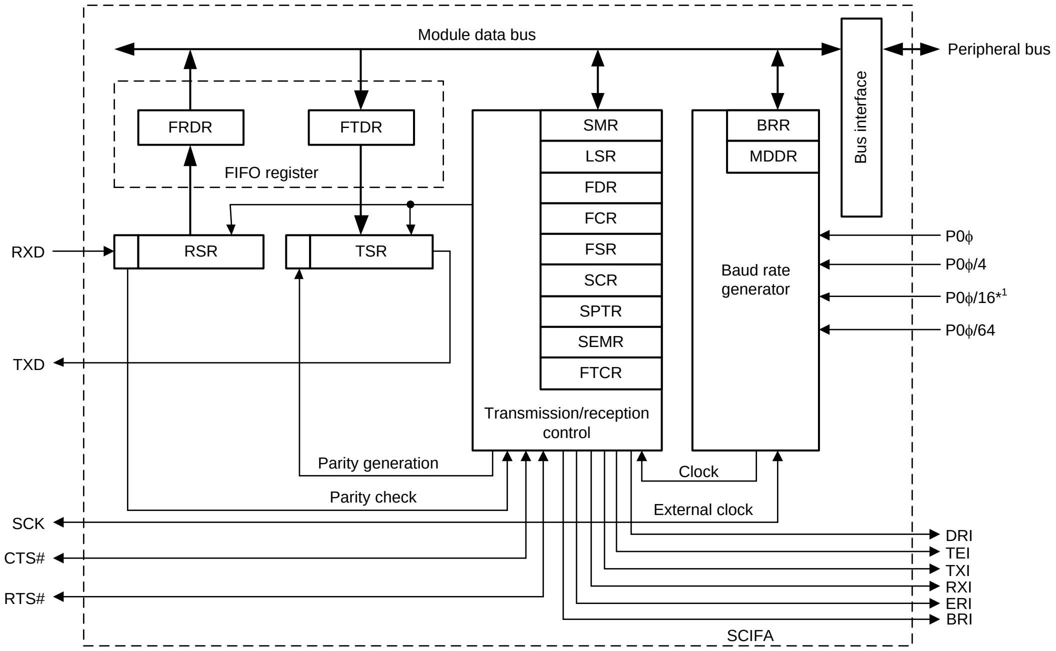
Table 22.1 lists the specifications of the SCIFA.

Table 22.1 Specifications of SCIFA

Item		Description
Channel		● 5 channels (ch 0,1,2,3,4) (RZ/G2L) ● 4 channels (ch 0,1,2,3) (RZ/G2LC)
Serial communication method		Asynchronous communication mode and clock synchronous communication mode
Transfer speed		Selectable bit rate with an on-chip baud rate generator
Full duplex communication		Transmitting section: realizes continuous data transmission using 16-stage FIFO buffer Receiving section: realizes continuous data reception using 16-stage FIFO buffer
Data transmission		Selectable either LSB-first or MSB-first transfer
Interrupt source		The following six sources: ● Transmit-end (TEIF) ● Transmit-FIFO-data-empty (TXIF) ● Receive-FIFO-data-full (RXIF) ● Receive-data-ready (DRIF)*1 ● Receive-error (ERIF) ● Break detection or overrun (BRIF)
Asynchronous communication mode	Character length	7 or 8 bits
	Transmission stop bit length	1 or 2 bits
	Parity	Even, odd, or none
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	Controls data transmission and reception using the CTS# and RTS# pins. Not applicable to channel 3 (SCIF3) and channel 4 (SCIF4)
	Break detection	Break signal detection function by hardware.
	Clock source	Selectable from internal or external clock
	Noise cancellation	Incorporates a digital noise filter in the RXD pin input path.
Clock synchronous communication mode	Character length	8 bits
	Receive error detection	Detects an overrun error as a receive error.
	Clock source	Selectable either internal or external clock
Bit rate modulation		Enables errors to be decreased by correcting the output of the on-chip baud rate generator.

Note 1. Effective only for asynchronous communication mode

Figure 22.1 shows a block diagram of the SCIFA.



RSR: Receive shift register
FRDR: Receive FIFO data register
TSR: Transmit shift register
FTDR: Transmit FIFO data register
SMR: Serial mode register
SCR: Serial control register
FSR: Serial status register
MDDR: Modulation duty register

BRR: Bit rate register
SPTR: Serial port register
FCR: FIFO control register
FDR: FIFO data count register
LSR: Line status register
SEMR: Serial extended mode register
FTCR: FIFO trigger control register

Note 1. The P0ϕ/16 clock is not input in the case of the channel 0 interface.

Figure 22.1 Block Diagram of SCIFA

Table 22.2 lists the input/output pins of the SCIFA.

Table 22.2 Pin Configuration of the SCIFA

Channel	Item	Pin Name	I/O	Function
0 to 4 (n = 0-4)	Serial clock pin	SCIFn_SCK	I/O	Transmission/reception clock input/output, general output
	Receive data pin	SCIFn_RXD	Input	Receive data input
	Transmit data pin	SCIFn_TXD	Output	Transmit data output
0 to 2 (n = 0-2)	Transmission/reception start control pin	SCIFn_CTS#	I/O	Input for hardware flow control (transmission enable signal) / general output
		SCIFn_RTS#	Output	Output for hardware flow control (transmission request signal) / general output

Note: Channels of each pin is omitted.

22.2 Register Descriptions

Table 22.3 Base Address Table

Address Space	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Cortex-A55 Address Space	H’0_1004_B800	H’0_1004_BC00	H’0_1004_C000	H’0_1004_C400	H’0_1004_C800
Cortex-M33 Address Space Non-Secure	H’4004_B800	H’4004_BC00	H’4004_C000	H’4004_C400	H’4004_C800
Cortex-M33 Address Space Secure	H’5004_B800	H’5004_BC00	H’5004_C000	H’5004_C400	H’5004_C800

Table 22.4 List of Registers

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Serial mode register	SMR	R/W	H’0000	H’00	16
Bit rate register	BRR	R/W	H’FF	H’02	8
Modulation duty register	MDDR	R/W	H’FF	H’02	8
Serial control register	SCR	R/W	H’0000	H’04	16
Transmit FIFO data register	FTDR	W	Undefined	H’06	8
Serial status register	FSR	R/W	H’0020	H’08	16
Receive FIFO data register	FRDR	R	Undefined	H’0A	8
FIFO control register	FCR	R/W	H’0000	H’0C	16
FIFO data count register	FDR	R	H’0000	H’0E	16
Serial port register	SPTR	R/W	H’00xx	H’10	16
Line status register	LSR	R/W	H’0000	H’12	16
Serial extended mode register	SEMR	R/W	H’00	H’14	8
FIFO trigger control register	FTCR	R/W	H’1F1F	H’16	16

Note: BRR and MDDR are located in the same address. Setting the MDDRS bit of the SEMR register switches these registers.

22.2.1 Receive Shift Register (RSR)

The RSR register receives serial data and temporally stores the data. The SCIFA stores the serial data input via the RXD pin into the RSR register and converts the data to the parallel form. When one byte of data has been received, it is automatically transferred to the receive FIFO data register (FRDR).

The CPU cannot read from or write to the RSR register directly.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0

22.2.2 Receive FIFO Data Register (FRDR)

The FRDR register is a 8-bit, 16-stage FIFO register that stores the received serial data. When the SCIFA receives one byte of serial data, it transfers the received data from the receive shift register (RSR) to the FRDR register and completes the receive operation. Continuous reception is possible until the received 16 bytes of data are stored. If the FRDR register is read when there is no received data in the FRDR register, an undefined value is read.

When the FRDR register is full of received data, subsequently received serial data is lost.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0

22.2.3 Transmit Shift Register (TSR)

The SCIFA transfers the transmit data from the transmit FIFO data register (FTDR) to the TSR register, and then transmits the data serially to the TXD pin. After transmitting one byte of data, the SCIFA automatically transfers the next transmit data from the FTDR register into the TSR register and starts transmission.

The CPU cannot read from or write to the TSR register directly.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1

22.2.4 Transmit FIFO Data Register (FTDR)

The FTDR register is a 8-bit, 16-stage FIFO register that stores serial transmission data. When the SCIFA detects that the transmit shift register (TSR) is empty, it transmits data written in the FTDR register to the TSR register and starts serial transmission. Continuous serial transmission is executed until there is no transmit data left in the FTDR register. Writing the transmit data to the FTDR register should be done when a transmit data empty interrupt (TXI) request is generated. When the FTDR register becomes full of transmit data (16 bytes), no more data can be written. Even if new data is written, the data is ignored.

CPU can read from the FTDR register but cannot write to it.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1

22.2.5 Serial Mode Register (SMR)

The SMR register specifies the SCIFA serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to the SMR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CM	CHR	PE	PM	STOP	—	CKS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKS[1:0]	All 0	R/W	Clock Select b1 b0 0 0: 1 × P0ϕ[*1] 0 1: 1/4 × P0ϕ[*1] 1 0: 1/16 × P0ϕ[*1,*2] 1 1: 1/64 × P0ϕ[*1]
b2	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b3	STOP	0	R/W	Stop Bit Length 0: One stop bit 1: Two stop bits
b4	PM	0	R/W	Parity Mode 0: Even parity 1: Odd parity
b5	PE	0	R/W	Parity Enable 0: Parity bit addition or check is disabled. 1: Parity bit addition or check is enabled.
b6	CHR	0	R/W	Character Length 0: 8-bit data 1: 7-bit data[*3]
b7	CM	0	R/W	Communication Mode 0: Asynchronous mode 1: Clock synchronous mode
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. P0ϕ: SCIFn_CLK_PCK (n = 0 to 4)

Note 2. This setting is not applicable to channel 0. It is only applicable to channels 1 to 4.

Note 3. When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.

CKS[1:0] Bits (Clock Select)

Select an internal clock source for the on-chip baud rate generator. For further information on the clock source, bit rate register settings, and baud rates, see **Section 22.2.8, Bit Rate Register (BRR)**.

STOP Bit (Stop Bit Length)

Selects one bit or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character.

Note: When transmitting with one stop bit, a single 1 bit (stop bit) is added at the end of each transmission character.

Note: When transmitting with two stop bits, two 1 bits (stop bits) are added at the end of each transmission character.

PM Bit (Parity Mode)

Selects either the even or odd parity check. The setting of this bit is effective only when the parity enable (PE) bit of this register is set to 1 in asynchronous mode. The setting of this bit is ignored in clock synchronous mode, or when parity addition/check is disabled in asynchronous mode.

Note: If even parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s even in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is even.

Note: If odd parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s odd in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is odd.

PE Bit (Parity Enable)

Selects whether to add a parity bit on data transmission and whether to enable/disable the parity check on data reception in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the setting of this bit.

Note: When this bit is set to 1, an even or odd parity bit specified in the PM bit is added to data to be transmitted. The SCIFA verifies whether the parity bit of the received data is even or odd as specified in the PM bit when receiving.

CHR Bit (Character Length)

Selects 7- or 8-bit data length in asynchronous mode. In clock synchronous mode, the data length is always 8 bits, regardless of the CHR setting.

CM Bit (Communication Mode)

Selects whether the SCIFA operates in asynchronous or clock synchronous mode.

22.2.6 Serial Control Register (SCR)

The SCR register enables or disables the SCIFA transmission/reception and interrupt requests, and selects the transmit/receive clock source. The CPU can always read from and write to the SCR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	TEIE	CKE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKE[1:0]	All 0	R/W	Clock Enable In asynchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for input pin (input signal is ignored). The SCK pin state depends on the SCKIO and SCKDT bits in SPTR. 0 1: Internal clock or SCK pin is used for clock output (The output clock frequency is 16 or 8 times of the bit rate). 1 0: External clock or SCK pin is used for clock input (The input clock frequency is 16 or 8 times of the bit rate). 1 1: Setting prohibited In clock synchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for synchronous clock output. 0 1: Internal clock or SCK pin is used for synchronous clock output. 1 0: External clock or SCK pin is used for synchronous clock input. 1 1: Setting prohibited
b2	TEIE*[1]	0	R/W	Transmit End Interrupt Enable 0: Transmit end interrupt (TEI) request is disabled. 1: Transmit end interrupt (TEI) request is enabled.
b3	REIE	0	R/W	Receive Error Interrupt Enable 0: Receive-error interrupt (ERI) and break interrupt (BRI) requests are disabled. 1: Receive-error interrupt (ERI) and break interrupt (BRI) requests are enabled.
b4	RE	0	R/W	Receive Enable 0: Data reception is disabled. 1: Data reception is enabled.
b5	TE	0	R/W	Transmit Enable 0: Data transmission is disabled. 1: Data transmission is enabled.
b6	RIE	0	R/W	Receive Interrupt Enable 0: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled. 1: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled.
b7	TIE	0	R/W	Transmit Interrupt Enable 0: Transmit-FIFO-data-empty interrupt request (TXI) is disabled. 1: Transmit-FIFO-data-empty interrupt request (TXI) is enabled.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. TEI interrupt requests can be cleared by reading 1 from the TEND flag, and then clearing the setting to 0, or by setting the TEIE bit to 0.

CKE[1:0] Bits (Clock Enable)

Select the SCIFA clock source and enable or disable clock output from the SCK pin. Depending on the settings of these bits, the SCK pin can be used for serial clock output or serial clock input. If the SCK pin is set for the synchronous clock output in the clock synchronous mode, set the CM bit in the SMR register to 1, and then set the CKE[1:0] bits. The settings of the CKE[1:0] bits are listed in **Table 22.17**.

REIE Bit (Receive Error Interrupt Enable)

Specifies whether to enable or disable a receive-error interrupt (ERI) request and a break interrupt (BRI) request. The setting of this bit is only valid when the RIE bit is set to 0.

Note: ERI interrupt requests can be cleared by reading 1 from the ER bit in the FSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0. BRI interrupt requests can be cleared by reading 1 from the BRK bit in the FSR register, or from the ORER flag in the LSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0.

RE Bit (Receive Enable)

Specifies whether to enable or disable the serial data reception.

Note: Setting this bit to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, and PER in the FSR register, and ORER in the LSR register). These flags retain their previous values.

Note: Serial reception starts when a start bit is detected in asynchronous mode, or a synchronous clock input is detected in clock synchronous mode. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the receive format and reset the receive FIFO.

TE Bit (Transmit Enable)

Specifies whether to enable or disable the serial data transmission.

Note: Serial transmission starts after writing of data to be transmitted into the FTDR register under this condition. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the transmit format and reset the transmit FIFO.

RIE Bit (Receive Interrupt Enable)

Specifies whether to enable or disable a receive-FIFO-data-full (RXI) interrupt request when the RDF flag in the serial status register (FSR) is set to 1, a receive-data ready (DRI) interrupt request when the DR flag in the FSR register is set to 1, a receive-error (ERI) interrupt request when the ER flag in the FSR register is set to 1, and a break (BRI) interrupt request when the BRK flag in the FSR register or the ORER flag in the line status register (LSR) is set to 1.

Note: RXI interrupt requests can be cleared by reading 1 from the DR or RDF flag in the FSR register, then clearing the flag to 0, or by clearing the RIE bit to 0. DRI interrupt requests can be cleared by reading 1 from the DR flag in the FSR register, and then clearing the setting to 0, or by clearing the RIE bit in this register to 0. Receive error interrupt (ERI) requests and break interrupt (BRI) requests can be cleared by clearing both the RIE and REIE bits in this register to 0.

TIE Bit (Transmit Interrupt Enable)

Specifies whether to enable or disable a transmit-FIFO-data-empty interrupt (TXI) request when the serial transmit data is transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the transmit FIFO data register falls below the specified trigger number for transmission, and the TDFE flag in the serial status register (FSR) is set to 1.

Note: TXI interrupt requests can be cleared either by writing a greater quantity of transmit data than the specified transmission trigger number into the FTDR register, reading 1 from the TDFE flag, and then clearing the TDFE flag to 0, or by clearing this bit to 0.

22.2.7 Serial Status Register (FSR)

The FSR register is a 16-bit register. The 8 lower-order bits indicate the status flag representing the SCIFA operating state.

The CPU can always read and write to the FSR register, but cannot write 1 to the status flags (ER, TEND, TDFE, BRK, RDF, and DR bits) in this register. These flags can be only cleared to 0 when they have first been read (after being set to 1). b3 (FER) and b2 (PER) are read-only bits that cannot be written.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R	R	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
b0	DR	0	R/(W)*1	Receive Data Ready Flag 0: Reception is in progress, or no received data has remained in the FRDR register after normally completed receiving. 1: Next receive data has not been received.
b1	RDF	0	R/(W)*1	Receive FIFO Data Full Flag 0: The quantity of receive data in the FRDR register falls below the specified reception trigger number. 1: The quantity of receive data written in the FRDR register is equal to or greater than the specified reception trigger number.
b2	PER	0	R	Parity Error Flag*4 0: No receive parity error occurred in the next receive data read from the FRDR register. 1: A receive parity error occurred in the next receive data read from the FRDR register.
b3	FER	0	R	Framing Error Flag*4 0: No receive framing error occurred in the next data read from the FRDR register. 1: A receive framing error occurred in the next data read from the FRDR register.
b4	BRK	0	R/(W)*1	Break Detect Flag 0: No break signal is received. 1: A break signal is received.*2
b5	TDFE	1	R/(W)*1	Transmit FIFO Data Empty Flag 0: The quantity of transmit data written in the FTDR register exceeds the specified transmission trigger number. 1: The quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.*3
b6	TEND	0	R/(W)*1	Transmit End Flag 0: Transmission is in the waiting state or in progress. 1: Transmission is completed.
b7	ER	0	R/(W)*1	Receive Error Flag 0: Reception is in progress or has normally completed. 1: A framing error or parity error has occurred during reception.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. 0 can be only written to clear the flag after 1 is read.

Note 2. When a break signal is detected, transfer of the receive data (H'00) to the FRDR register stops after the detection. When the break ends and the receive signal becomes mark state (high level), the transfer of receive data resumes.

Note 3. Since the FTDR register is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is “16 minus the number of non-transmitted data units”. If additional data is written, the data is ignored. The quantity of data in the FTDR register is indicated by the 8 higher-order bits of the FDR register.

Note 4. When the DMAC is used to read data, the generation of errors cannot be checked by reading this flag.

DR Bit (Receive Data Ready Flag)

Indicates that the quantity of data stored in the receive FIFO data register (FRDR) falls below the specified reception trigger number, and that no next data has been received yet after the elapse of 15 ETUs from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1.

[Setting condition]

- DR is set to 1 when the FRDR register contains less data than the specified reception trigger number, and no next data has been received yet after the elapse of 15 ETUs*[1] from the last stop bit.

[Clearing conditions]

When either of the following is satisfied:

- DR is cleared to 0 when DR = 1 is read and then 0 is written to the DR flag.
- DR is cleared to 0 when all received data in the FRDR register are read.

Note 1. This is equivalent to one and a half (1.5) frames in the 8-bit format with one stop bit (ETU: elementary time unit).

Note: When the RE bit in SCR is cleared to 0, the DR bit is not affected and retains its previous value.

RDF Bit (Receive FIFO Data Full Flag)

Indicates that receive data has been transferred to the receive FIFO data register (FRDR), and the quantity of data in FRDR becomes equal to or greater than the specified reception trigger number.

[Setting condition]

- RDF is set to 1 when the quantity of receive data which is equal to or greater than the specified reception trigger number are stored in the FRDR register*[1].

[Clearing conditions]

- RDF is cleared to 0 when RDF = 1 is read and then 0 is written to this bit.
- RDF is cleared to 0 when the FRDR register is read.

Note 1. Since the FRDR register is a 16-byte FIFO register, the maximum quantity of data that can be read when this bit is 1 is equivalent to the specified reception trigger number. If an attempt is made to read after all the data in the FRDR register has been read, the read data is undefined. The quantity of receive data in the FRDR register is indicated by the 8 lower-order bits of the FDR register.

PER Bit (Parity Error Flag)

Indicates whether there is a parity error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- PER is set to 1 when a parity error is present in the next data read from the FRDR register.

[Clearing condition]

- PER is cleared to 0 when no parity error is present in the next data read from the FRDR register.

FER Bit (Framing Error Flag)

Indicates whether there is a framing error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- FER is set to 1 when a framing error is present in the next data read from the FRDR register.

[Clearing condition]

- FER is cleared to 0 when no framing error is present in the next data read from the FRDR register.

BRK Bit (Break Detect Flag)

Indicates that a break signal has been detected in receive data.

[Setting condition]

- BRK is set to 1 when data including a framing error is received, and the framing error is followed by at least one frame of data received at the space 0 level (low level).

[Clearing condition]

- BRK is cleared to 0 when software reads BRK after it has been set to 1 and then writes 0 to BRK.

TDFE Bit (Transmit FIFO Data Empty Flag)

Indicates that data has been transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the FRDR register becomes equal to or less than the specified transmission trigger number, and writing of transmit data to the FRDR register is enabled.

[Setting conditions]

When either of the following is satisfied:

- TDFE is set to 1 when the TE bit in SCR is 0.
- TDFE is set to 1 when the quantity of transmit data written in the FRDR register is equal to or less than the specified transmission trigger number.

[Clearing conditions]

- TDFE is cleared to 0 when 0 is written in the TDFE bit after reading TDFE = 1.
- When transmit data is written to the FTDR register

TEND Bit (Transmit End Flag)

Indicates that the FRDR register contains no more valid data and transmission is completed when transmitting the last bit of the transmit data.

[Setting condition]

When the following is satisfied:

- TEND is set to 1 when the FTDR register does not contain transmit data when the last bit of the serial transmission data is transmitted.

[Clearing conditions]

- When transmit data is written to the FTDR register
- When 0 is written to TEND after it has been read as 1

ER Bit (Receive Error Flag)

Indicates the occurrence of a framing error, or of a parity error when receiving the parity-added data*[1].

[Setting conditions]

When either of the following is satisfied:

- ER is set to 1 when the stop bit is found to be 0 after checking whether the stop bit of the received data is 1 at the end of one data receive operation*[1].
- ER is set to 1 when the total number of 1s in the received data and parity bit combined does not match the even or odd parity setting specified by the PM bit in the SMR register.

[Clearing condition]

- When 0 is written to ER after it has been read as 1

Note 1. Clearing the RE bit to 0 in the SCR register does not affect this bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to the FRDR register and the receive operation is continued. Whether the data read from the FRDR register includes a receive error can be detected by the FER and PER bits in the FSR register.

Note: In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.

22.2.8 Bit Rate Register (BRR)

The BRR register is an 8-bit register that, together with the baud rate generator clock source selected by the CKS[1:0] bits in the serial mode register (SMR), determines the serial transmit/receive bit rate.

This register is located in the same address as that of the MDDR register and selected when the MDDRS bit in SEMR is 0. The CPU can read and write to BRR. Writing to BRR should be executed when TE = RE = 0 in the SCR register.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The BRR setting is calculated using the following formulae.

[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$N = \frac{P0\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$N = \frac{P0\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$N = \frac{P0\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$N = \frac{P0\phi}{16 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

[Clock synchronous mode]

$$N = \frac{P0\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)
N: Setting of the BRR register ($0 \leq N \leq 255$) (The setting must satisfy the electrical characteristics).
P0ϕ: Operating frequency for peripheral modules (MHz)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see **Table 22.5**).

NOTE

The MDDR register is used to adjust the bit rate. For details, see **Section 22.2.9, Modulation Duty Register (MDDR)**.

Table 22.5 SMR Register Setting

n	Clock Source	SMR Register Settings	
		CKS1	CKS0
0	P0ϕ	0	0
1	P0ϕ/4	0	1
2	P0ϕ/16	1	0
3	P0ϕ/64	1	1

The bit rate error in asynchronous mode is calculated using the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 64 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 32 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 32 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0))

$$\text{Error (\%)} = \left\{\frac{\text{P0}\phi \times 10^6}{(\text{N}+1) \times \text{B} \times 16 \times 2^{2\text{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1))

Table 22.6 list the examples of the BRR register setting in asynchronous mode, and **Table 22.7** list the examples of the BRR register setting in clock synchronous mode.

Table 22.6 Bit Rates and BRR Register Settings in Asynchronous Mode

Bit Rate (bps)	P0ϕ (MHz)		
	100		
	n	N	Error (%)
150	—	—	—
300	3	162	−0.15
600	2	80	0.47
1200	2	162	−0.15
2400	2	80	0.47
4800	1	162	−0.15
9600	1	80	0.47
14400	0	216	0.01
19200	0	162	−0.15
28800	0	108	−0.45
31250	0	99	0.00
38400	0	80	0.47
115200	0	26	0.47
500000	0	5	*1

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0.
When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled.
When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled.
Configure settings so the range of error is no greater than 1%.

Note 1. Values for the blank cells in the table can be set using the MDDR register.
For details, see **Section 22.2.9, Modulation Duty Register (MDDR)** and the **Table 22.12**.

Table 22.7 Bit Rates and BRR Register Settings in Clock Synchronous Mode

Bit Rate (bps)	P0ϕ (MHz)	
	100	
	n	N
250	—	—
500	—	—
1000	—	—
2500	3	155
5000	3	77
10000	3	38
25000	1	249
50000	1	124
100000	0	249
250000	0	99
500000	0	49
1000000	0	24
2500000	0	9
5000000	0	4

Note: Continuous transmission or reception is not possible.
Set the BRR register so that the range of error can fall within 1% or less.

Remarks: —: Setting is prohibited.

Table 22.8 lists the maximum bit rates for various frequencies in asynchronous mode when the baud rate generator is used. **Table 22.9** lists the maximum bit rates for various frequencies in clock synchronous mode when the baud rate generator is used. **Table 22.10** and **Table 22.11** list the maximum rates for external clock inputs in asynchronous mode and clock synchronous mode, respectively.

Table 22.8 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Asynchronous Mode)

P0ϕ (MHz)	Maximum Bit Rate (bit/s)	Settings	
		n	N
100	12500000	0	0

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 1. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is 1/2. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is 1/4.

Table 22.9 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Clock Synchronous Mode)

P0ϕ (MHz)	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
	Maximum Bit Rate (bit/s)	Settings		Maximum Bit Rate (bit/s)	Settings	
		n	N		n	N
100	25000000	0	0	12500000	0	1

Table 22.10 Maximum Bit Rates with External Clock Input (in Asynchronous Mode)

P0ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
100	25.00	3125000

Note: This is an example when the SEMR.ABCS0 bit is 1. When the ABCS0 bit is set to 0, the bit rate is 1/2.

Table 22.11 Maximum Bit Rates with External Clock Input (in Clock Synchronous Mode)

P0ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
100	8.33	8330000

22.2.9 Modulation Duty Register (MDDR)

The MDDR register corrects the bit rate adjusted by the BRR register. The value after reset of this register is H’FF. When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (MDDR/256). The relationship between the MDDR register setting and the bit rate (B) is given by the following formula. The MDDR register is located in the same address as that of the BRR register and is selected when the MDDRS bit in SEMR is 1. This register is only writable when TE = RE = 0 in the SCR register. b7 in this register is fixed to 1.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The formulae below show the relationships between the MDDR setting and the bit rate (B) when the bit rate modulation function is used.

[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$B = \frac{P0\phi \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{P0\phi \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$B = \frac{P0\phi \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{P0\phi \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

[Clock synchronous mode]

$$B = \frac{P0\phi \times 10^6}{8 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

When the bit rate modulation is used, the bit rate average error is given by the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\mathrm{Error}\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\mathrm{Error}\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\mathrm{Error}\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\mathrm{Error}\ (\%) = \left\{\frac{P0\phi \times 10^6}{B \times 16 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

B: Bit rate (bits/s)
N: BRR register setting ($0 \leq N \leq 255$)
The setting must satisfy the electrical characteristics).
P0ϕ: Operating frequency for peripheral modules (MHz)
MDDR: MDDR setting ($128 \leq MDDR \leq 256$)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see **Table 22.5**).

Table 22.12 Bit Rates and BRR and MDDR Registers Settings in Asynchronous Mode

	P0ϕ (MHz)			
	100			
Bit Rate (bps)	n	N	MDDR	Error (%)
150	3	252	199	0.02
300	3	161	255	0.08
600	3	80	255	0.08
1200	2	161	255	0.08
2400	2	80	255	0.08
4800	1	161	255	0.08
9600	1	80	255	0.08
14400	—	—	—	—
19200	0	161	255	0.08
28800	0	107	254	0.08
31250	—	—	—	—
38400	0	80	253	0.08
115200	0	26	255	0.08
500000	0	5	246	0.10

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.

22.2.10 FIFO Control Register (FCR)

The FCR register resets the quantity of data in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR) and specifies the number of triggers. This register also specifies whether to enable the loop-back test.

The CPU can always read and write to the FCR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	LOOP	0	R/W	Loop-Back Test 0: Loop back test is disabled. 1: Loop back test is enabled
b1	RFRST	0	R/W	Receive FIFO Data Register Reset 0: Normal operation 1: Resets the FRDR register.
b2	TFRST	0	R/W	Transmit FIFO Data Register Reset 0: Normal operation 1: Resets the FTDR register.
b3	MCE	0	R/W	Modem Control Enable 0: Model signal is disabled.*1 1: Model signal is enabled.
b5, b4	TTRG[1:0]	All 0	R/W	Transmit FIFO Data Trigger Number Select b5 b4 0 0: 8 (8)*2 0 1: 4 (12)*2 1 0: 2 (14)*2 1 1: 0 (16)*2
b7, b6	RTRG[1:0]	All 0	R/W	Receive FIFO Data Trigger Number Select In asynchronous mode: b7 b6 0 0: 1 0 1: 4 1 0: 8 1 1: 14 In clock synchronous mode: b7 b6 0 0: 1 0 1: 2 1 0: 8 1 1: 14
b10 to b8	RSTRG[2:0]	All 0	R/W	RTS# Output Active Trigger Number Select b10 b8 0 0 0: 15 0 0 1: 1 0 1 0: 4 0 1 1: 6 1 0 0: 8 1 0 1: 10 1 1 0: 12 1 1 1: 14

Bit	Bit Name	Initial Value	R/W	Description
b15 to b11	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. The CTS# input level does not affect the transmit operation. Similarly, the RTS# input level does not affect the receive operation.

Note 2. Values in parentheses mean the number of empty bytes in the FTDR register when the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

LOOP Bit (Loop-Back Test)

Internally connects between the transmit output pin (TXD) and the receive input pin (RXD) and between the RTS# pin and the CTS# pin, to perform loop-back testing.

RFRST Bit (Receive FIFO Data Register Reset)

Disables the receive data in the receive FIFO data register (FRDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

TFRST Bit (Transmit FIFO Data Register Reset)

Disables the transmit data in the transmit FIFO data register (FTDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

MCE Bit (Modem Control Enable)

Specifies whether to enable or disable the modem control signals, CTS# and RTS#. In clock synchronous mode, this bit should always be set to 0.

TTRG[1:0] Bits (Transmit FIFO Data Trigger Number Select)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR). When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

The setting in these bits is valid when the TTRGS bit in the FTCR register is 0. When the TTRGS bit in the FTCR register is 1, the setting of the TFTC[4:0] bits in the FTCR register is valid.

RTRG[1:0] Bits (Receive FIFO Data Trigger Number Select)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR). When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXI) request is generated.

The setting in these bits is valid when the RTRGS bit in the FTCR register is 0. When the RTRGS bit in the FTCR register is 1, the setting of the RFTC[4:0] bits in the FTCR register is valid.

RSTRG[2:0] Bits (RTS# Output Active Trigger Number Select)

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number, the RTS# signal is in the high state.

The setting in these bits is only valid when a modem signal is enabled by the MCE bit in this register in asynchronous mode.

22.2.11 FIFO Data Count Register (FDR)

The FDR register indicates the quantity of data stored in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR).

This register indicates the quantity of transmit data in the FTDR register with the 8 higher-order bits, and the quantity of receive data in the FRDR register with the 8 lower-order bits. The CPU can always read the FDR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	T[4:0]					—	—	—	R[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	R[4:0]	All 0	R	Receive Data Quantity in FRDR Indicate the quantity of receive data stored in the FRDR register.
b7 to b5	—	All 0	R	Reserved These bits are read as 0.
b12 to b8	T[4:0]	All 0	R	Non-Transmitted Data Quantity in FTDR Indicate the quantity of non-transmitted data stored in the FTDR register.
b15 to b13	—	All 0	R	Reserved These bits are read as 0.

R[4:0] Bits

Indicate the quantity of receive data stored in the FRDR register.

H’00 means no received data, and H’10 means that all of the received data is stored in the FRDR register.

T[4:0] Bits

Indicate the quantity of non-transmitted data stored in the FTDR register.

H’00 means no transmit data, and H’10 means that all of the data for transmission is stored in the FTDR register.

22.2.12 Serial Port Register (SPTR)

The SPTR register controls input/output and data of the pins multiplexed to SCIFA function. The CPU can always read and write to the SPTR register.

NOTE

b6, b4, b2, and b0 of this register respectively indicate the input status of their corresponding pins. See the descriptions for each bit for details. Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RTS2IO	RTS2DT	CTS2IO	CTS2DT	SCKIO	SCKDT	SPB2IO	SPB2DT
Initial Value	0	0	0	0	0	0	0	0	0	x	0	x	0	x	0	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Remarks: x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
b0	SPB2DT	x	R/W	Serial Port Break Data Select Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2IO bit. See **Table 22.15**.
b1	SPB2IO	0	R/W	Serial Port Break Input/Output Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.
b2	SCKDT	x	R/W	SCK Port Data Select Controls the SCK pin in combination with the CM bit in the SMR register, the SCKIO bit, and the CKE1 and CKE0 bits in the SCR register. See **Table 22.17**.
b3	SCKIO	0	R/W	SCK Port Input/Output Controls the SCK pin in combination with the CM bit in the SMR register, the SCKDT bit, and the CKE1 and CKE0 bits in the SCR register. See **Table 22.17**.
b4	CTS2DT	x	R/W	CTS# Port Data Select Controls the CTS# pin in combination with MCE bit in FCR and CTS2IO bit. See **Table 22.14**. The SCIF 3 and SCIF 4 channels are not supported.
b5	CTS2IO	0	R/W	CTS# Port Output Specify Controls the CTS# pin in combination with MCE bit in FCR and CTS2IO bit. See **Table 22.14**. The SCIF 3 and SCIF 4 channels are not supported.
b6	RTS2DT	x	R/W	RTS# Port Data Select Controls the RTS# pin in combination with MCE bit in FCR and RTS2IO bit. See **Table 22.13**. The SCIF 3 and SCIF 4 channels are not supported.
b7	RTS2IO	0	R/W	RTS# Port Output Specify Controls the RTS# pin in combination with MCE bit in FCR and RTS2IO bit. See **Table 22.13**. The SCIF 3 and SCIF 4 channels are not supported.
b15 to b8	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Remarks: x: Undefined

SPB2DT Bit (Serial Port Break Data Select)

This bit specifies the output level of the TXD pin when the setting of the SCR.TE bit is 0. The RXD pin input status can be read from this bit regardless of the SPB2IO bit setting. However, the RXD pin function must have been selected with the general I/O port.

SPB2IO Bit (Serial Port Break Input/Output)

Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.

SCKDT Bit (SCK Port Data Select)

The SCK pin status can be read from this bit regardless of the SCKIO bit setting. (When the SCK pin is used for input, the input signal is invalid (has no means) but the pin status can be read.) However, the SCK pin function must have been selected with the general I/O port.

SCKIO Bit (SCK Port Input/Output)

Specifies input or output status of the SCK pin. This bit controls the SCK pin in combination with the SCKDT bit, the CM bit in the SMR register, and the CKE1 and CKE0 bits in the SCR register.

CTS2DT Bit (CTS# Port Data Select)

The status of the CTS# pin can be read from this bit regardless of the CTS2IO bit setting. However, the CTS# pin function must have been selected with the general I/O port.

RTS2DT Bit (RTS# Port Data Select)

The status of the RTS# pin can be read from this bit regardless of the RTS2IO bit setting. However, the RTS# pin function must have been selected with the general I/O port

Table 22.13 RTS# Pin Status

FCR.MCE Bit Setting	RTS2IO Bit Setting	RTS2DT Bit Setting	RTS# Pin Status
0	0	×	Setting prohibited*[1]
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control output

Remarks: ×: Don’t care

Note 1. There is no problem with the initial setting if the RTS# pin is not used.

Table 22.14 CTS# Pin Status

FCR.MCE Bit Setting	CTS2IO Bit Setting	CTS2DT Bit Setting	CTS# Pin Status
0	0	×	Setting prohibited*[1]
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control input

Remarks: ×: Don’t care

Note 1. There is no problem with the initial setting if the CTS# pin is not used.

Table 22.15 TXD Pin Status

SCR.TE Bit Setting	SPB2IO Bit Setting	SPB2DT Bit Setting	TXD Pin Status
0	0	×	Setting prohibited
0	1	0	Low output
0	1	1	High output
1	×	×	Transmit data output

Remarks: ×: Don’t care

22.2.13 Line Status Register (LSR)

The LSR register is a 16-bit register. The PER and FER bits indicate the number of receive errors in the receive FIFO data register. 1 cannot be written to the OREER status flag. The flag should be read as 1 prior to clearing it to 0.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PER[3:0]				—	—	FER[3:0]				—	ORER
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R(W)*1

Bit	Bit Name	Initial Value	R/W	Description
b0	ORER	0	R/(W) *1	Overrun Error Flag 0: Reception is in progress or has normally completed. 1: An overrun error has occurred during reception.
b1	—	0	R	Reserved This bit is read as 0.
b5 to b2	FER[3:0]	All 0	R	Framing Error Count Indicates the quantity of data with a framing error among the receive data stored in the receive FIFO data register (FRDR).
b7, b6	—	All 0	R	Reserved These bits are read as 0.
b11 to b8	PER[3:0]	All 0	R	Parity Error Count Indicates the quantity of data with a parity error among the receive data stored in the receive FIFO data register (FRDR).
b15 to b12	—	All 0	R	Reserved These bits are read as 0.

Note 1. To clear the flag, 0 can be only written after 1 is read.

ORER Bit (Overrun Error Flag)

Indicates that receive operation abnormally stops due to occurrence of an overrun error. This flag is not affected and retains its previous state if the RE bit in the serial control register (SCR) is cleared to 0. The receive FIFO data register (FRDR) retains the data before an overrun error occurred, and newly received data is lost. When the ORER bit is set to 1, the SCIFA cannot continue subsequent serial reception.

[Setting condition]

- When the next serial reception is completed with the receive FIFO in full state (16-byte data is received)

[Clearing condition]

- When 0 is written to ORER after being read as 1.

Note: When the internal clock is selected while the SCIFA is in clock synchronous mode, the amount of receive data can be controlled, so no overrun occurs.

FER[3:0] Bits (Framing Error Count)

The values of bits 5 to 2 indicate the quantity of data with a framing error after the ER bit in the FSR register is set. Reading 0000 from the FER[3:0] bits means all 16-byte receive data in the FRDR register have a framing error.

PER[3:0] Bits (Parity Error Count)

The values of bits 11 to 8 indicate the quantity of data with a parity error after the ER bit in the FSR register is set. Reading 0000 from the PER[3:0] bits means all 16-byte receive data in the FRDR register have a parity error.

22.2.14 Serial Extended Mode Register (SEMR)

The SEMR register specifies either LSB or MSB first, enables the noise cancellation, operation in normal or double-speed mode of the baud rate generator, and bit rate modulation, and selects the modulation register and the sampling count (either 8 or 16 times).

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	BGDM	—	BRME	MDDRS	DIR	NFEN	—	ABCS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	ABCS0	0	R/W	Asynchronous Base Clock Select 0: Operates on a frequency 16 times the transfer rate as the base clock. 1: Operates on a frequency 8 times the transfer rate as the base clock.
b1	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b2	NFEN	0	R/W	Noise Cancellation Enable 0: Noise cancellation for the RxD pin is disabled. 1: Noise cancellation for the RxD pin is enabled.
b3	DIR	0	R/W	Data Transfer Direction Select 0: Transmits the data in the FTDR register by the LSB-first method. The received data is stored in the FRDR register by the LSB- first method. 1: Transmits the data in the FTDR register by the MSB-first method. The received data is stored in the FRDR register by the MSB- first method.
b4	MDDRS	0	R/W	Modulation Duty Register Select 0: BRR register is accessible. 1: MDDR register is accessible.
b5	BRME	0	R/W	Bit Rate Modulation Enable 0: Bit rate modulation is disabled. 1: Bit rate modulation is enabled.
b6	—	0	R/W	Reserved This bit is read as 0. The write value should be 0.
b7	BGDM	0	R/W	Baud Rate Generator Double- Speed Mode Select 0: Baud rate generator normal mode: Baud rate generator operates on the clock signal produced by dividing the clock source by two. 1: Baud rate generator double-speed mode: Baud rate generator operates on the clock signal produced by the clock source (no frequency division).

ABCS0 Bit (Asynchronous Base Clock Select)

Selects the base clock for 1-bit period in asynchronous mode.

This bit setting is valid only in asynchronous mode (i.e., when the CM bit in the SMR register is 0).

NFEN Bit (Noise Cancellation Enable)

Reduces noise of the input to the RxD pin. This function is only valid in asynchronous mode. For details, see **Section 22.7, Noise Cancellation**.

In clock synchronous mode, this bit should always be set to 0.

DIR Bit (Data Transfer Direction Select)

Selects the serial communication format. This bit is valid only when the transmit/receive data is in 8-bit formats.*[1]

Note 1. Asynchronous mode or clock synchronous mode with the 8-bit data length

MDDRS Bit (Modulation Duty Register Select)

Selects the register to be enabled access to it.

BRME Bit (Bit Rate Modulation Enable)

Specifies whether to enable or disable the bit rate modulation.

BGDM Bit (Baud Rate Generator Double-Speed Mode Select)

Selects operating mode of the baud rate generator. When setting 1 in this bit, the baud rate generator included in the SCIFA operates in double-speed mode. The setting of this bit is only effective in asynchronous mode (SMR.CM bit = 0) when the internal clock is selected as the clock source (SCR.CKE[1:0] = 00b). Use normal mode under any other settings.

22.2.15 FIFO Trigger Control Register (FTCR)

The FTCR register is a 16-bit register that specifies FIFO trigger conditions. The CPU can always read from and write to the FTCR register.

Bit	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RTRGS	—	—	RFTC[4:0]					TTRGS	—	—	TFTC[4:0]				
Initial Value	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
b4 to b0	TFTC[4:0]	All 1	R/W	Transmit FIFO Data Trigger Number H’00: Transmit data trigger number is 0. H’0F: Transmit data trigger number is 15. Do not set H’10 to H’1F in these bits.
b6, b5	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b7	TTRGS	0	R/W	Transmit Trigger Select 0: TTRG[1:0] bits in FCR are valid. 1: TFTC[4:0] bits in FTCR are valid.
b12 to b8	RFTC[4:0]	All 1	R/W	Receive FIFO Data Trigger Number H’01: Receive data trigger number is 1. H’10: Receive data trigger number is 16. Do not set H’00 and H’11 to H’1F in these bits.
b14, b13	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.
b15	RTRGS	0	R/W	Receive Trigger Select 0: RTRG[1:0] bits in FCR are valid. 1: RFTC[4:0] bits in FTCR are valid.

TFTC[4:0] Bits (Transmit FIFO Data Trigger Number)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR).

When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXI) request is generated.

RFTC[4:0] Bits (Receive FIFO Data Trigger Number)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR).

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXI) request is generated.

22.3 Operation

22.3.1 Overview

For serial communication, the SCIFA can select either asynchronous mode in which characters are synchronized individually or a clock synchronous mode in which communication is synchronized with clock pulses.

The SCIFA has a 16-stage FIFO buffer for both transmission and receptions, reducing the overhead of the CPU and enabling continuous high-speed communication. The RTS# and CTS# signals are provided as modem control signals. Selection of a transmission/reception format is enabled with the serial mode register (SMR). **Table 22.16** shows the transmission format which can be selected in the serial mode register (SMR). As shown in **Table 22.17**, the SCIFA clock source can be selected by the combination of the CKE1 and CKE0 bits in the serial control register (SCR).

(1) Asynchronous Mode

- Data length is selectable either 7 or 8 bits
- Parity addition and 1- or 2-bit stop bit addition are selectable.
  (The combination of the preceding selections determines the transmission/reception format and character length).
- In reception, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The stored data quantities are indicated in the FIFO data count register (FDR), respectively for transmit and receive FIFO data.
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of on-chip baud rate generator and can output the clock with a frequency 16 (or 8) times the bit rate.
  When an external clock is selected, the external clock input must have a frequency 16 (or 8) times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- The transmission/reception format is fixed to the 8-bit data length.
- In reception, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of the on-chip baud rate generator, and outputs this clock to external devices as the synchronous clock.
  When an external clock is selected, the SCIFA operates on the input synchronous clock not using the on-chip baud rate generator.

Table 22.16 SMR Register Settings and SCIFA Communication Formats

SMR Register					SCIFA Transmission/Reception Format		
b7	b6	b5	b3				
CM	CHR	PE	STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous mode	8 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	×	×	×	Clock synchronous mode	8 bits	Not set	None

Remarks: ×: Don’t care

Table 22.17 SMR, SCR, and SPTR Register Settings and SCIFA Clock Source Selection

SMR Register	SCR Register		SPTR Register				
b7	b1	b0	b3	b2			
CM	CKE1	CKE0	SCKIO	SCKDT	Mode	Clock Source	SCK Pin Function
0	0	0	0	×	Asynchronous mode	Internal	Input pin (input signal invalid) (Initial state)
			1	0			SCK pin state: Low
			1	1			SCK pin state: High
		1	×	×			Outputs a clock with frequency 16/8 times the bit rate*1
	1	0	×	×		External	Inputs a clock with frequency 16/8 times the bit rate*2
		1	×	×		Setting prohibited	
1	0	×	×	×	Clock synchronous mode	Internal	Outputs the synchronous clock
	1	0	×	×		External	Inputs the synchronous clock
		1	×	×		Setting prohibited	

Remarks: ×: Don’t care

Note 1. SEMR.ABCS0 = 0: Output a clock that has a frequency 16 times the bit rate. SEMR.ABCS0 = 1: Output a clock that has a frequency 8 times the bit rate.

Note 2. SEMR.ABCS0 = 0: Input a clock that has a frequency 16 times the bit rate. SEMR.ABCS0 = 1: Input a clock that has a frequency 8 times the bit rate.

22.3.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIFA are independent, so full duplex communication is possible. The transmitter and receiver are 16-stage FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmission and reception.

Figure 22.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIFA monitors the line and starts serial communication when the line goes to the space (low) state, considered as a start bit. One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIFA synchronizes at the falling edge of the start bit. The SCIFA samples each data bit on the eighth pulse of a clock with a frequency 16 or 8 times the bit rate*[1]. Receive data is latched at the center of each bit.

Note 1. When the SEMR.ABCS0 bit = 0, data is sampled on the eighth pulse of a clock with a frequency 16 times the bit rate.
When the SEMR.ABCS0 bit = 1, data is sampled on the fourth pulse of a clock with a frequency 8 times the bit rate.

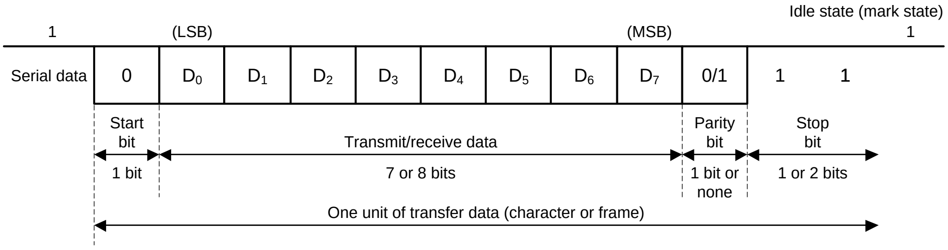


Figure 22.2 Data Format in Asynchronous Communication
(8-Bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)

(1) Transmit/Receive Formats

Table 22.18 lists the eight communications formats that can be selected in asynchronous mode. The format is selected by setting in the serial mode register (SMR).

Table 22.18 Serial Communications Formats (in Asynchronous Mode)

SMR Setting			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
		1	START	8-bit data								STOP	STOP	
	1	0	START	8-bit data								P	STOP	
		1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
		1	START	7-bit data							STOP	STOP		
	1	0	START	7-bit data							P	STOP		
		1	START	7-bit data							P	STOP	STOP	

Note: START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

An SCIFA transmit/receive clock can be selected from two types of clock sources: the internal clock generated by the on- chip baud rate generator, the external clock input from the SCK pin. The clock source is selected by the settings of the CM bit in the serial mode register (SMR), the CKE[1:0] bits in the serial control register (SCR), and the ACS0 bit in the serial extended mode register (SEMR). For clock source selection, refer to **Table 22.17**.

When an external clock is input at the SCK pin, it must have a frequency equal to 16/8 times the desired bit rate.

When the SCIFA operates on an internal clock, it can output a clock signal on the SCK pin. The frequency of this output clock is 16/8 times the desired bit rate.

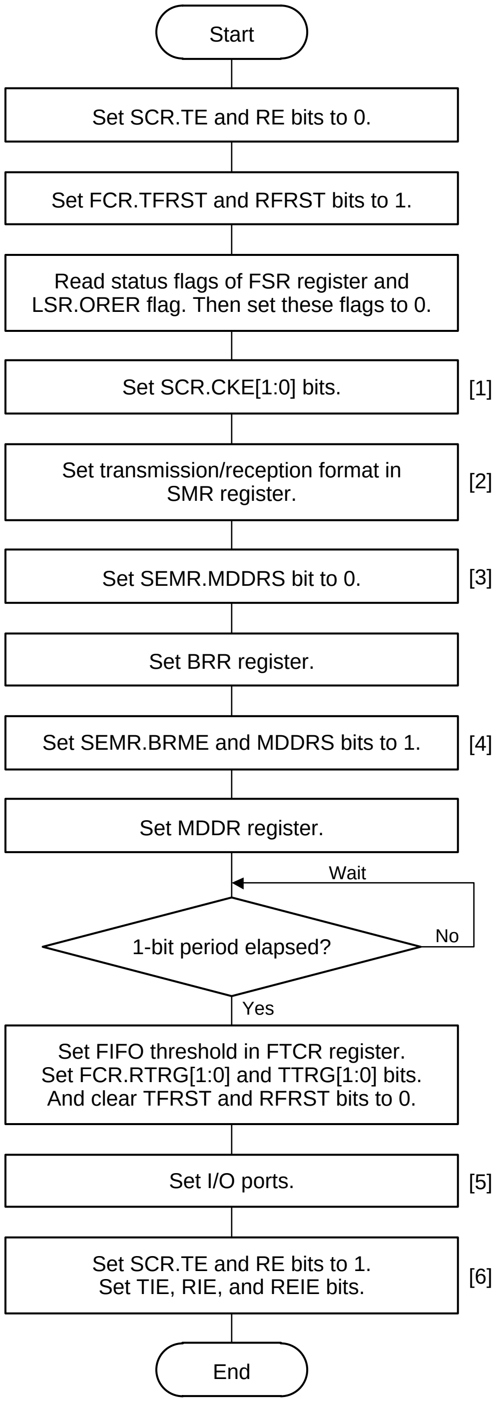
(3) Transmitting and Receiving Data

SCIFA Initialization (in Asynchronous Mode)

Before transmitting or receiving data, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA as follows.

When changing operating mode or communication format, always clear the TE and RE bits in the SCR register to 0 before following the procedure given below. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing TE and RE to 0, however, does not initialize the serial status register (FSR), transmit FIFO data register (FTDR), or receive FIFO data register (FRDR), which retain their previous contents. Clear TE to 0 after all transmit data has been transmitted and the TEND flag in the FSR register is set. The TE bit can be cleared to 0 during transmission, but the transmit data (the TXD pin output level) after the TE bit is cleared to 0 depends on the settings of the SPB2IO and SPB2DT bits in the SPTR register. Set the TFRST bit in the FCR register to 1 and reset the FTDR register before TE is set to 1 again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIFA operation becomes unreliable if the clock is stopped. **Figure 22.3** shows a sample flowchart for initializing the SCIFA in asynchronous mode.



[1] Set the TIE, RIE, TE, and RE bits in the SCR register to 0 before setting the CKE[1:0] bits.

[2] Set the transmission/reception format in the SMR register.

[3] Clear the SEMR.MDDRS bit to 0 and write a value corresponding to the bit rate in the BRR register. Not necessary if an external clock is used.

[4] Set the SEMR.BRME and MDDRS bits to 1 and write a value corresponding to the bit rate in the MDDR register. Not necessary if an external clock is used. Wait for at least 1-bit period before proceeding to the next step.

[5] Set I/O ports and the MPC register so that they can be used as the SCK, TXD, and RXD pins.

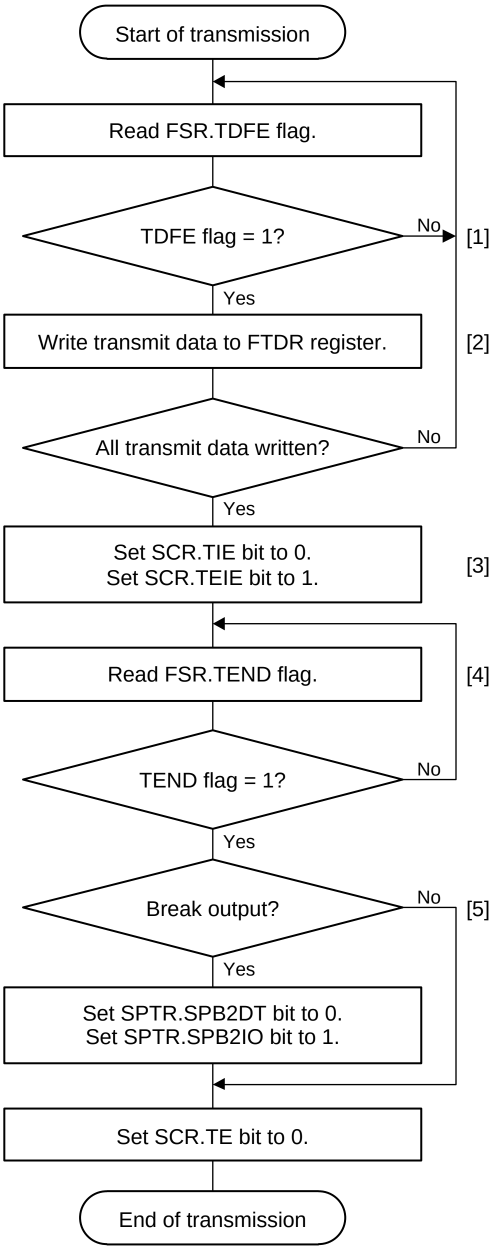
[6] After setting the MDDR register, wait for at least 1-bit period and then write 1 to the SCR.TE or RE bit. To permit interrupt requests, set TIE, RIE and REIE bits to 1. Setting the TE and RE bits to 1 makes the TXD and RXD pins available to be used. The TXD pin enters the mark state and the RXD pin enters the idle state waiting for a start bit.

Figure 22.3 Sample Flowchart for SCIFA Initialization in Asynchronous Mode

Transmitting Serial Data (in Asynchronous Mode)

Figure 22.4 shows a sample flowchart for serial transmission in asynchronous mode.

Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data to the FTDR register. The transmit data quantity that can be written is “16 - FDR.T[4:0]”. A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure:
When transmit data is written to the FTDR register, the TDFE flag is automatically cleared to 0.

[3] Serial transmission end procedure (when TXIF or TEIF interrupt is used):
After the last transmission data is written to the FTDR register, set the TIE bit to 0 and the TEIE bit to 1 in the SCR register. If the DMAC is activated to write the transmit data to the FTDR register, enable the transmit end interrupt. This will generate a CPU interrupt request when the last transmit data is written in the FTDR register. The TIE and TEIE bit settings will be changed by the CPU interrupt process.
This procedure is unnecessary if the TXIF or TEIF interrupt is not used.

[4] Serial transmission end confirmation:
Read the FSR register and confirm that the TEND flag has changed to 1.
A change of the TEND flag from 0 to 1 can be known by a TEIF interrupt request.

[5] Break output during serial transmission:
To cause a break to be output during serial transmission, set the SPB2DT bit to 0 and the SPB2IO bit to 1 in the SPTR register. Then set the SCR.TE bit to 0.*[1]

In Step [1], the number of data bytes that can be written can be checked by the FDR.T[4:0] flag (non-transmitted data quantity in the FTDR register).

Note 1. The SPB2DT bit of the SPTR register indicates the input status of the RXD pin. Writings to the SPB2DT and SPB2IO bits of this register in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written to the SPB2DT bit. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Figure 22.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode

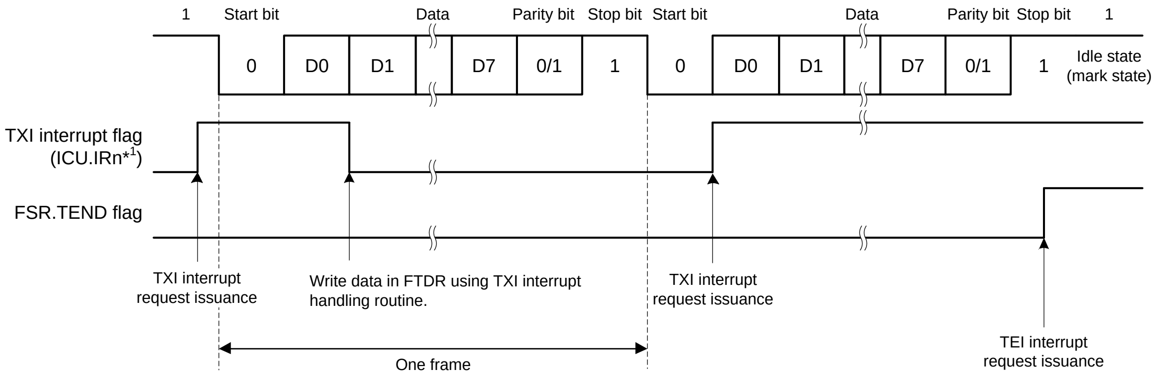
In asynchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXI interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number specified in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag is set. If the TIE bit in the serial control register (SR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated. The serial transmit data is output from the TXD pin in the following order.

a) Start bit:	One-bit 0 is output.
b) Transmit data:	8- or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
c) Parity bit:	One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
d) Stop bit(s):	One or two 1 bits (stop bits) are output.
e) Mark state:	1 is output continuously until the start bit that starts the next transmission is output.

3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the stop bit. If data is present, the data is transferred from the FTDR register to the TSR register, the stop bit is output, and then serial transmission of the next frame is started. If there is no data to be transmitted, the TEND flag in the FSR register is set to 1, the stop bit is output, and then the SCIFA enters the mark state (high level) in which 1 is output continuously.

Figure 22.5 shows an example of the operation for transmission in asynchronous mode.



Note 1. For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

Figure 22.5 Example of Transmit Operation in Asynchronous Mode
(8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

4. When modem control is enabled, transmission can be stopped/resumed by the input level to the CTS# pin. When a high level is input to the CTS# pin during transmission, the SCIFA enters the mark state (high level) after completion of one-frame data transmission. When a low level is input to the CTS# pin, output of the next data to be transmitted begins with a start bit. **Figure 22.6** shows an example of the operation for transmission when using the modem control function.

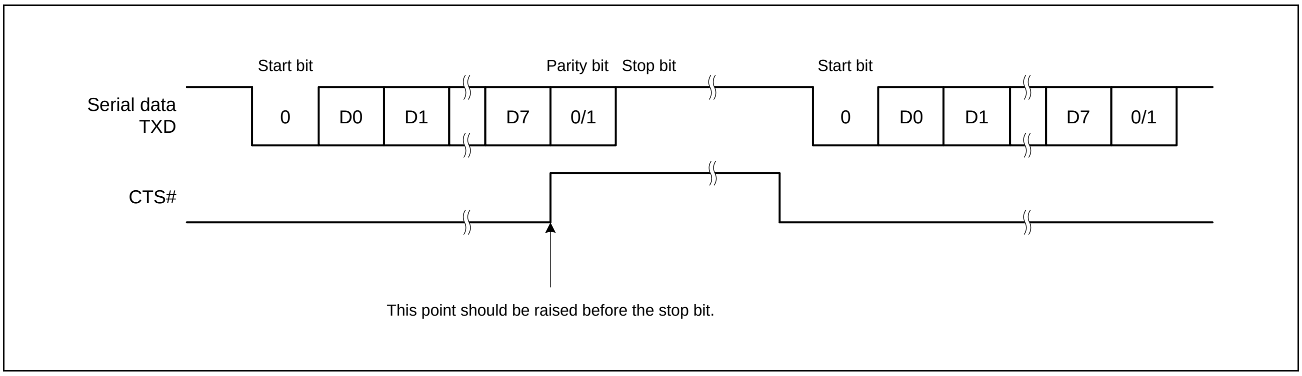
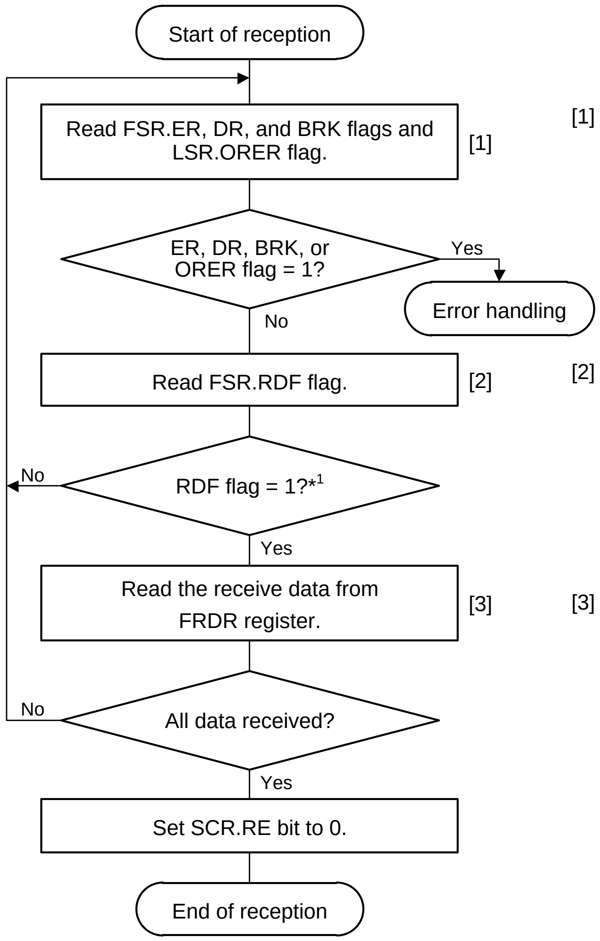


Figure 22.6 Example of Transmit Operation in Asynchronous Mode Using Modem Control Function (CTS#)

Receiving Serial Data (in Asynchronous Mode)

Figure 22.7 and **Figure 22.8** show sample flowcharts for serial reception in asynchronous mode. Follow the procedure given below for serial data reception after enabling the SCIFA for reception.



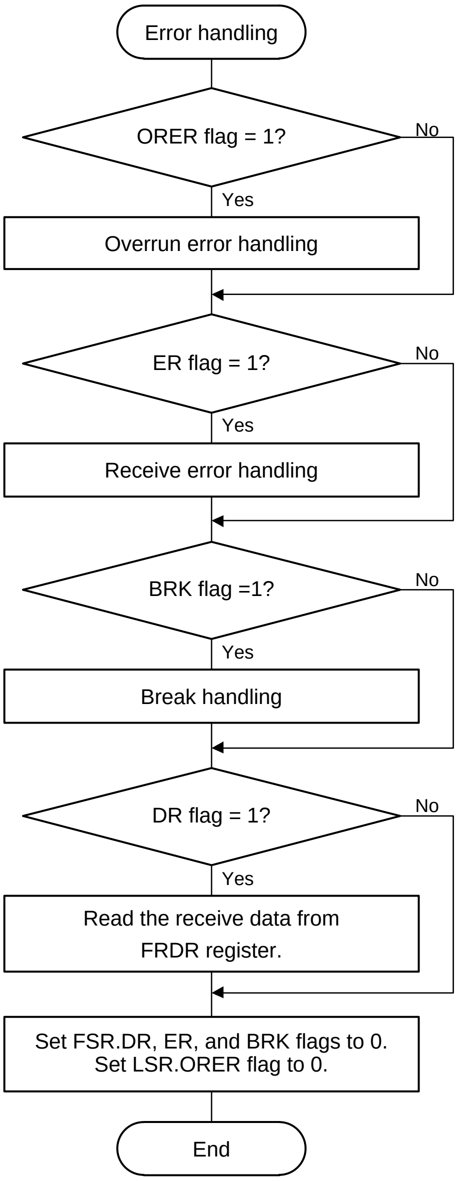
[1] Receive error handling and break detection:
When a receive error occurs, read the ER, DR, and BRK flags in the FSR register and the LSR.ORER flag to identify the error. After handling the error as specified, clear the ER, DR, BRK, and ORER flags to 0.
In the case where a framing error occurs, a break can be detected by reading the value of the RXD pin.
Error occurrence can be known by a BRIF, ERIF, or DRIF interrupt request.

[2] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data at least as much as the amount specified by the receive FIFO threshold from those stored in the FEDR register. The receive data quantity that can be read can be checked by the FDR.R[4:0] flag.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[3] Serial reception continuation procedure:
When receive data is read from the FRDR register, the RDF flag is automatically cleared to 0.

Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 22.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1)



- Whether a framing error or parity error has occurred in the received data that is read from the FRDR register can be checked with the FER and PER flags in the FSR register.
- When a break signal is received, the received data is not transferred to the FRDR register while the BRK flag is set to 1. However, note that the last data in the FRDR register is H'00 and the break data in which a framing error occurred is stored.

Figure 22.8 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (2)

In asynchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA monitors the communication line, and if a 0 start bit is detected, it performs internal synchronization to start reception.
2. The received data is stored into the RSR register in LSB-to-MSB order (when LSB-first transfer is selected).
3. The parity bit and stop bit are received.
   After receiving these bits, the SCIFA carries out the following checks.
   a) Stop bit check: The SCIFA checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   b) The SCIFA checks whether receive data can be transferred from the receive shift register (RSR) to the receive FIFO data register (FRDR).
   c) Parity bit check: The SCIFA checks whether the parity bit is an expected value.
   d) Overrun error check: The SCIFA checks whether the ORER flag is 0, indicating that the overrun error has not occurred.
   e) Break check: The SCIFA checks whether the BRK flag is 0, indicating that the break state is not set.

If all the above checks are passed, the receive data is stored in the FRDR register.

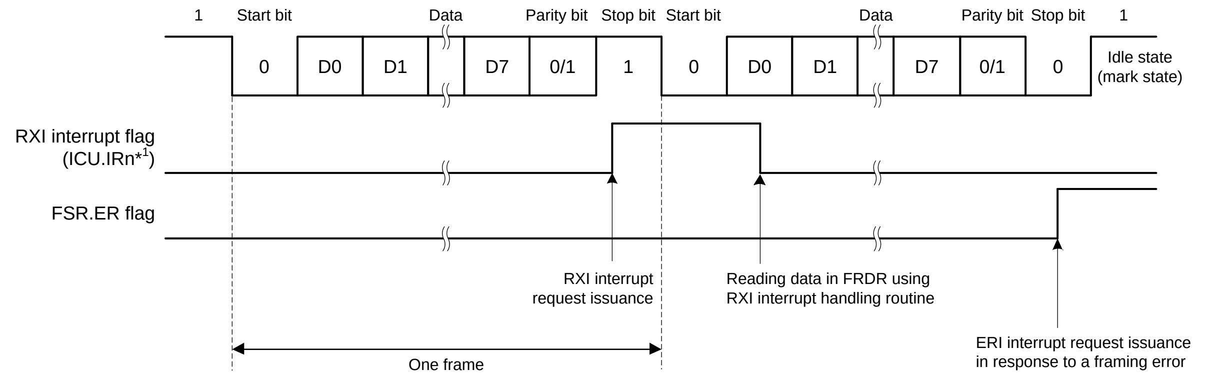
NOTE

When a parity error or a framing error occurs, reception is not suspended.

4. When receive data units equaling or exceeding the specified reception trigger number are stored in the receive FIFO data register (FRDR) and the RDF flag is changed to 1, a receive FIFO data full interrupt (RXI) request is generated while the RIE bit in the SCR register is set to 1. When the quantity of data in the FRDR register falls below the specified reception trigger number and the RIE bit in the SCR register is set to 1, a receive data ready interrupt (DRI) request is generated if no next data is received after the elapse of 15 ETUs*[1] from the last stop bit (the DR flag in the FSR register is 1). When the ER flag in the FSR register is changed to 1, a receive error interrupt (ERI) request is generated while the RIE or REIE bit in the SCR register is set to 1. When the BRK or ORER flag is changed to 1 in the FSR register, a break reception interrupt (BRI) request is generated while the RIE or REIE bit in the SCR register is set to 1.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

Figure 22.9 shows an example of the operation for reception in asynchronous mode.



Note 1. For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

Figure 22.9 Example of SCIFA Receive Operation in Asynchronous Mode
(8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

5. When modem control is enabled, the RTS# signal that indicates the FRDR register has space is output. When the RTS# pin is at low level, reception is possible. The RTS# pin being at the high level indicates that the number of entries in the FRDR register is equal to or greater than the threshold for output of the active level of the RTS# signal and that the transmission of further data needs to be suspended until the FRDR register has enough space. **Figure 22.10** shows an example of the operation for reception in asynchronous mode when using the modem control function.

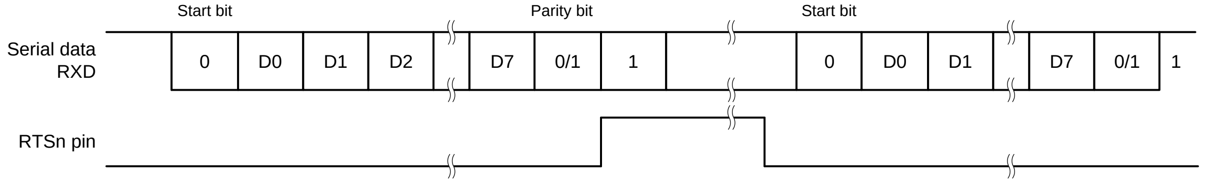


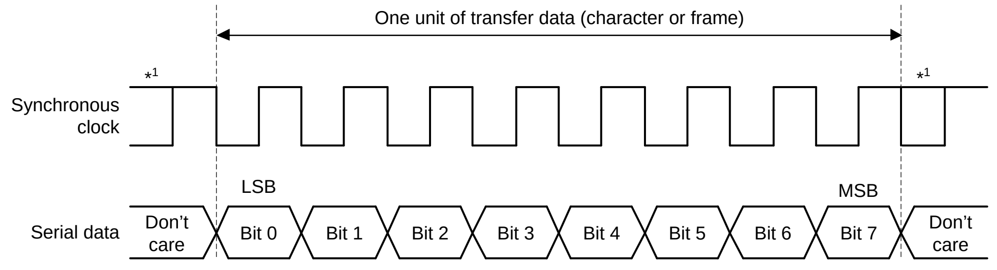
Figure 22.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Model Control Function (RTS#)

22.3.3 Operation in Clock Synchronous Mode

In clock synchronous mode, the SCIFA transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

Full-duplex communication is possible because the SCIFA transmitter and receiver are independent and share the same clock. Since the transmitter and the receiver have 16-stage FIFO buffers, respectively, continuous transmission or reception is possible by reading or writing data while transmission or reception is in progress.

Figure 22.11 shows the general format in clock synchronous serial communication.



Note 1. High level except in continuous transmission/reception

Figure 22.11 Data Format in Clock Synchronous Communication (when LSB-First Transfer is Selected)

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the synchronous clock to the next. Data is guaranteed valid at the rising edge of the synchronous clock.

In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB (when LSB-first transfer is selected).

In clock synchronous mode, the SCIFA receives data by synchronizing with the rising edge of the synchronous clock.

(1) Transmit/Receive Formats

The data length is fixed at eight bits.

No parity bit can be added.

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIFA transmit/receive clock according to the settings of the CM bit in the serial mode register (SMR) and the CKE[1:0] bits in the serial control register (SCR).

When the SCIFA operates on an internal clock, it outputs the synchronous clock signal at the SCK pin. Eight synchronous clock pulses are output per transmitted or received character. Unless the SCIFA is transmitting or receiving, the synchronous clock signal remains in the high state. When the SCIFA only receives data on an internal clock, the internal clock signal outputs while the RE bit in the SCR register is 1 until the number of data units in the receive FIFO reaches the specified reception trigger number.

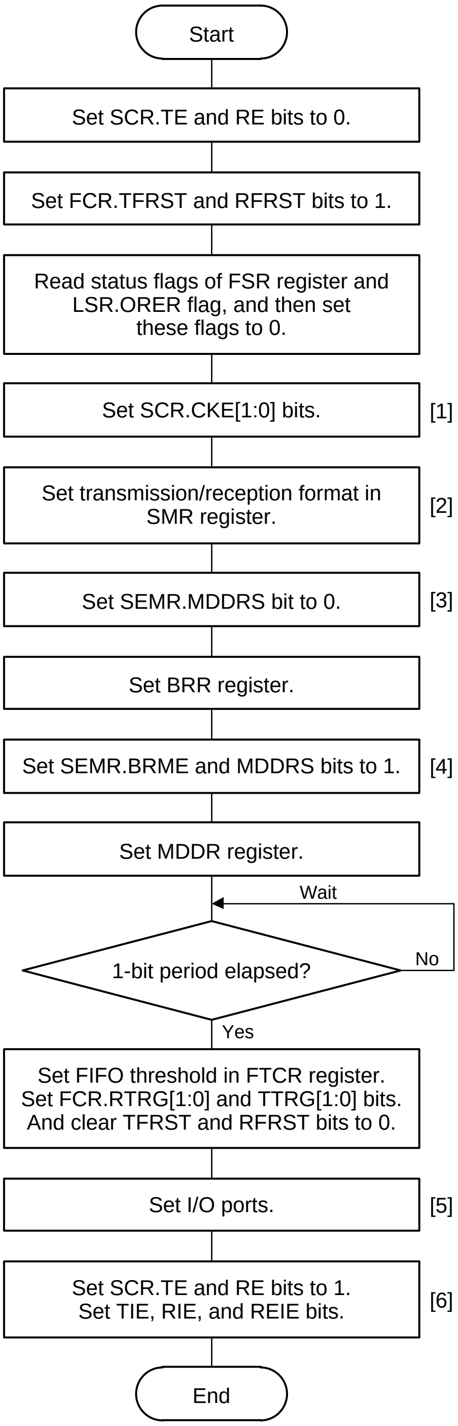
(3) Transmitting and Receiving Data

SCIFA Initialization (in Clock Synchronous Mode)

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA by performing the following procedure.

Similarly, before changing the mode or communication format, clear the TE and RE bits to 0, and then change it by performing the following procedure. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and the receive FIFO data register (FRDR), which retain their previous contents.

Figure 22.12 shows a sample flowchart for initializing the SCIFA in clock synchronous mode.



[1] Set the TIE, RIE, TE, and RE bits in the SCR register to 0 before setting the CKE[1:0] bits.

[2] Set the transmission/reception format in the SMR register.

[3] Clear the SEMR.MDDRS bit to 0 and write a value corresponding to the bit rate in the BRR register. Not necessary if an external clock is used.

[4] Set the SEMR.BRME and MDDRS bits to 1 and write a value corresponding to the bit rate in the MDDR register. Not necessary if an external clock is used. Wait for at least 1-bit period before proceeding to the next step.

[5] Set I/O ports and the MPC register so that they can be used as the SCK, TXD, and RXD pin.

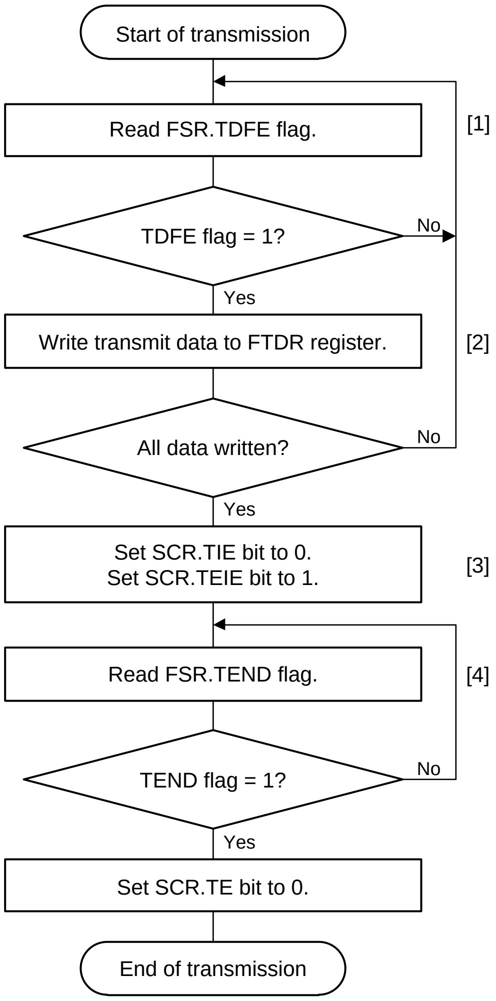
[6] After setting the MDDR register, wait for at least 1-bit, and then write 1 to the SCR.TE or RE bit. To permit interrupt requests, set TIE, RIE, and REIE bits to 1. Setting the TE and RE bits to 1 makes the TXD and RXD pins available to be used. In transmission, those pins enter the mark state, and in reception, they enter the idle state waiting for a start bit.

Figure 22.12 Sample Flowchart for SCIFA Initialization in Clock Synchronous Mode

Transmitting Serial Data (in Clock Synchronous Mode)

Figure 22.13 shows a sample flowchart for transmitting serial data in clock synchronous mode.

Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data to the FTDR register. The transmit data quantity that can be written is “16 - FDR.T[4:0]”.
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure:
When transmit data is written to the FTDR register, the TDFE flag is automatically cleared to 0.

[3] Serial transmission end procedure (when TXIF or TEIF interrupt is used):
After the last transmission data is written to the FTDR register, set the TIE bit to 0 and the TEIE bit to 1 in the SCR register. If the DMAC is activated to write the transmit data to the FTDR register, enable the transfer end interrupt. This will generate a CPU interrupt request when the last transmit data is written to the FTDR register. The TIE and TEIE bit settings will be changed by the CPU interrupt process.
This procedure is unnecessary if the TXIF or TEIF interrupt request is not used.

[4] Serial transmission end confirmation:
Read the FSR register and confirm that the TEND flag has changed to 1.
A change of the TEND flag from 0 to 1 can be known by a TEIF interrupt request.

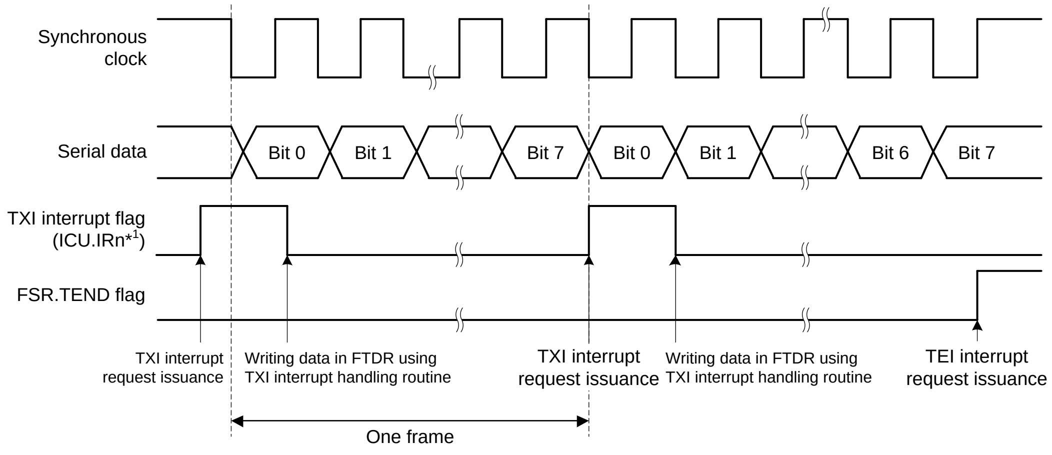
In Step [1], the number of data bytes that can be written can be checked by the FDR.T[4:0] flag (non-transmitted data quantity in the FTDR register).

Figure 22.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXI interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the specified number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number set in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag in the FSR register is set. If the TIE bit in the serial control register (SCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.
   If clock output mode is selected, the SCIFA outputs eight synchronous clock pulses. If an external clock source is selected, the SCIFA outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (b0) to the MSB (b7) (when LSB-first transfer is selected).
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the MSB (bit 7). If data is present, the data is transferred from the FTDR register to the TSR register, and then serial transmission of the next frame is started. If there is no data, the TXD pin holds the output level of the last data after the TEND flag in the FSR register is set to 1 and the MSB (bit 7) is output.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 22.14 shows an example of SCIFA transmit operation in clock synchronous mode.



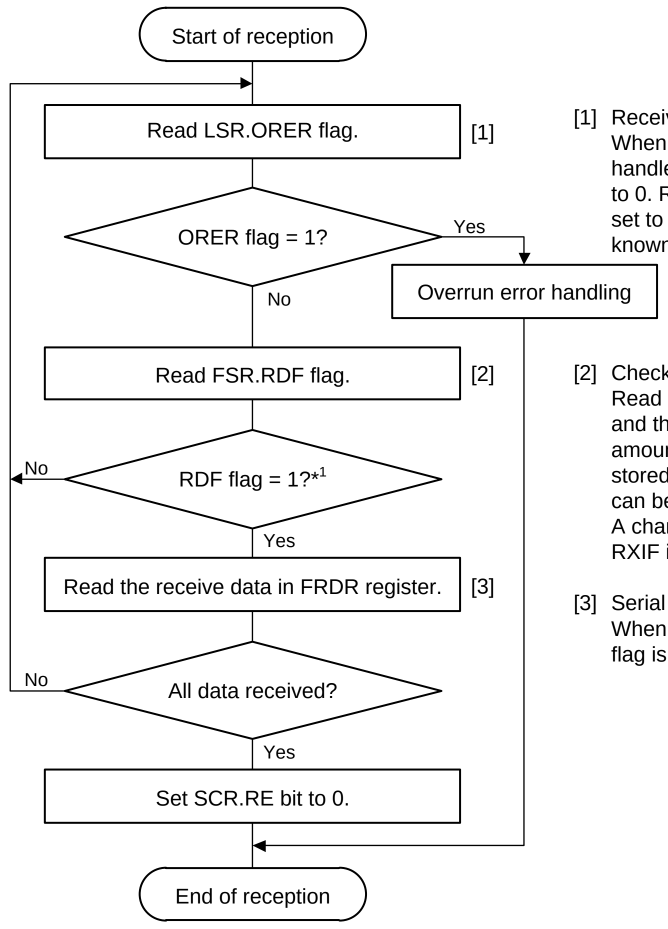
Note: For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

Figure 22.14 Example of SCIFA Transmit Operation in Clock Synchronous Mode (when LSB-First Transfer is Selected)

Receiving Serial Data (in Clock Synchronous Mode)

Figure 22.15 and **Figure 22.16** show sample flowcharts for receiving serial data in clock synchronous mode.

Follow the procedure given below for serial data reception after enabling the SCIFA for reception. When switching from asynchronous mode to clock synchronous mode without SCIFA initialization, make sure that the ORER, PER, and FER flags in the line status register (LSR) are cleared to 0.



[1] Receive error handling:
When a receive error occurs, read the LSR.ORER flag and handles the error as specified, and then clear the ORER flag to 0. Reception cannot be restarted while the ORER flag is set to 1. A change of the ORER flag from 0 to 1 can be known by a BRIF interrupt request.

[2] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data at least as much as the amount specified by the receive FIFO threshold from those stored in the FRDR register. The receive data quantity that can be read can be checked by the FDR.R[4:0] flag.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[3] Serial reception continuation procedure:
When receive data is read from the FRDR register, the RDF flag is automatically cleared to 0.

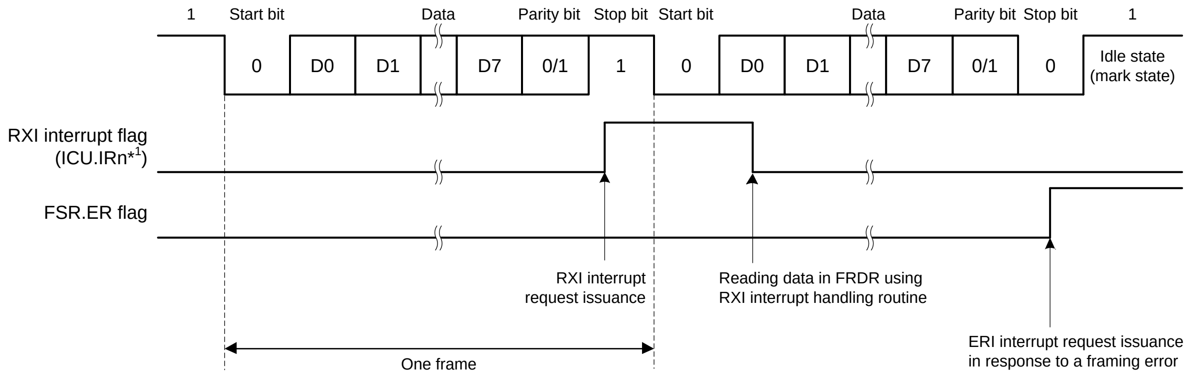
Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 22.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA synchronizes with the synchronous clock input or output and starts reception.
2. Receive data is stored into the receive shift register (RSR) in order from the LSB to the MSB (when LSB-first transfer is selected). After receiving the data, the SCIFA checks whether the receive data can be transferred from the RSR register to the FRDR register. If data can be transferred, the SCIFA stores the received data in the FRDR register. If an overrun error is detected during the error check, further reception is not performed.
3. After the received data units equaling or exceeding the specified reception trigger number are stored in the FRDR register and the RDF flag is set to 1, a receive-data-full interrupt (RXI) request is generated when the RIE bit in the serial control register (SCR) is set to 1. When the ORER flag in the line status register (LSR) is set to 1 and the RIE or REIE bit in the SCR register is also set to 1, a break interrupt (BRI) request is generated.

Figure 22.16 shows an example of SCIFA receive operation in clock synchronous mode.



Note 1. For the corresponding interrupt number, see **Section 8, Interrupt Controller**.

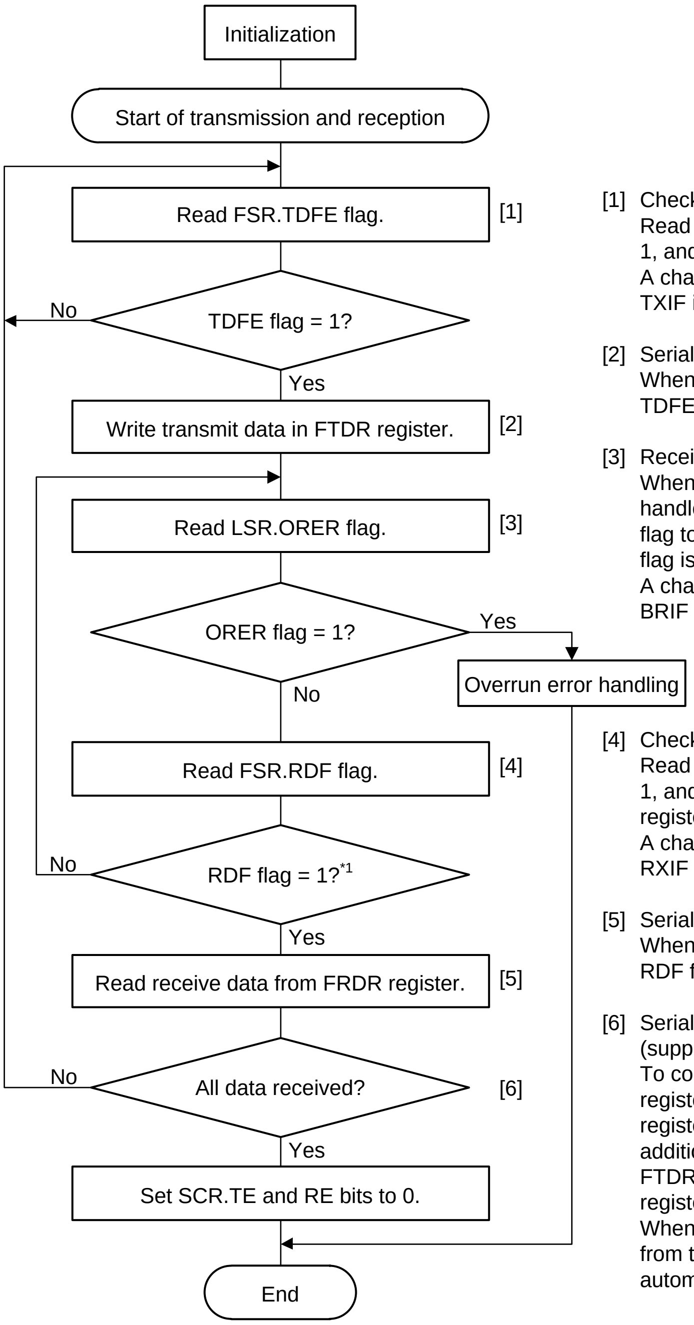
Figure 22.16 Example of SCIFA Receive Operation (when LSB-First Transfer is Selected)

Transmitting and Receiving Serial Data Simultaneously (in Clock Synchronous Mode)

Figure 22.17 shows a sample flowchart for transmitting and receiving serial data simultaneously in clock synchronous mode.

In simultaneous transmission/reception of serial data, number of receive data = number of transmit data = number of transmit data to be written to the FTDR register.

Follow the procedure given below for the simultaneous transmission/reception of serial data, after enabling the SCIFA for transmission/reception.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data in the FTDR register.
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure:
When transmit data is written to the FTDR register, the TDFE flag is automatically cleared to 0.

[3] Receive error handling:
When a receive error occurs, read the LSR.ORER flag and handles the error as specified, and then clear the ORER flag to 0. Reception cannot be restarted while the ORER flag is set to 1.
A change of the ORER flag from 0 to 1 can be known by a BRIF interrupt request.

[4] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data stored in the FRDR register.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[5] Serial reception continuation procedure:
When receive data is read from the FRDR register, the RDF flag is automatically cleared to 0.

[6] Serial transmission/reception continuation procedure (supplementary):
To continue serial transmission/reception, read the FRDR register and clear the RDF flag to 0 before the FRDR register overflows and the ORER flag changes to 1. In addition, before the last bit of the last data written to the FTDR register is transmitted, write the data in the FTDR register and clear the TDFE flag to 0.
When writing data to the FTDR register or reading data from the FRDR register, the TDFE flag and RDF flag are automatically cleared to 0.

Note: When switching the operation from transmission or reception to the simultaneous transmission and reception, clear the TE and RE bits to 0 and then simultaneously set these bits to 1.

Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 22.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode

22.4 Bit Rate Modulation

Using the bit rate modulation, the bit rate can be corrected by skipping the specified number of clock pulses input to the baud rate generator. To correct the bit rate, only the number of clock pulses specified in the MDDR register are enabled among 256 internal clock pulses specified by the CKS1 and CKS0 bits in the SMR register in a way that forms average intervals.

Figure 22.18 shows an example where P0φ is selected by the CKS[1:0] bits in SMR and BRR and MDDR are set to 0 and 160, respectively, in asynchronous mode. In this example, the cycle of the base clock is evenly corrected (256/160) and the bit rate is also corrected (160/256). Note that skipping an internal clock causes bias and expansion or contraction is generated in the pulse width of the base clock.

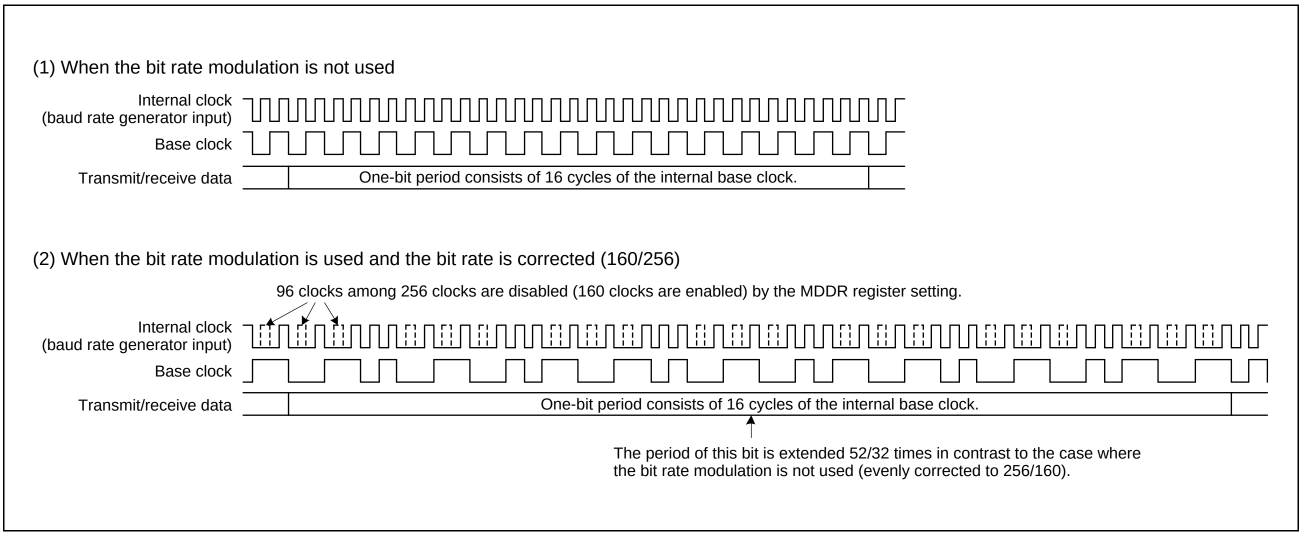


Figure 22.18 Example of Internal Base Clock when Bit Modulation is Used

22.5 Interrupt Sources

The SCIFA has six interrupt sources: transmit-FIFO-data-empty (TXI), receive-error (ERI), receive-FIFO-data-full (RXI), break (BRI), transmit-end (TEI), receive-data-ready (DRI). The TEI, DRI, ERI, and BRI interrupts share the same vector number.

Table 22.19 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the TIE, RIE, REIE, TEIE bits in the SCR register and are separately input to the interrupt controller.

When the quantity of transmit data written in the FTDR register as a result of transmission is equal to or less than the specified transmission trigger number, the TDFE flag in the serial status register (FSR) is set to 1 and a TXI interrupt request is generated.

When the data units equaling or exceeding the specified transmission trigger number are stored in the receive FIFO register (FRDR) and the RDF flag in the FSR register is set to 1, a receive data full interrupt (RXI) request is generated. When the quantity of received data in the FRDR register is below the specified reception trigger number and no next data has been received yet even after the period of 15 ETUs elapsed*[1] from the last stop bit, the DR flag in the FSR register is set to 1 and a receive data ready interrupt (DRI) request is generated. In clock synchronous mode, a DRI interrupt request is not generated.

When the BRK flag in the FSR register or the ORER flag in the LSR register is set to 1, a BRI interrupt request is issued. When the ER flag in the FSR register is set to 1, an ERI interrupt request is issued.

When the TEND flag in the FSR register is set to 1, a TEI interrupt request is issued.

When the RIE bit is cleared to 0 and the REIE bit in the SCR register is set to 1, an ERI and a BRI interrupt requests are issued but an RXI interrupt request is not.

An TXI interrupt indicates that transmit data can be written and an RXI interrupt indicates that receive data is stored in the FRDR register.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

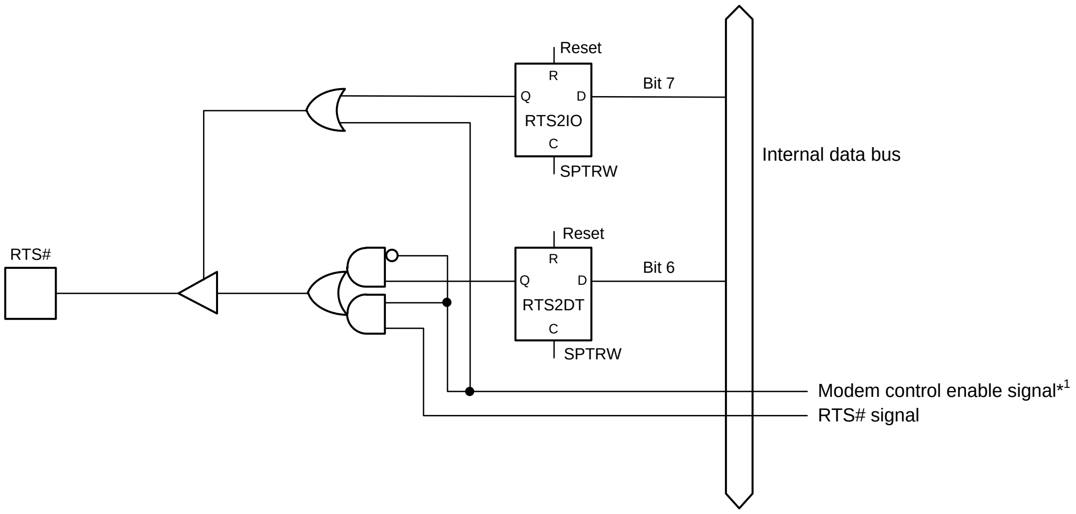
Table 22.19 SCIFA Interrupt Sources

Name	Level/Edge	Interrupt Source	Interrupt Enable Bit	DMAC Activation	Priority
BRI	Level	Interrupt caused by break (BRK) or overrun (ORER).	RIE or REIE	Impossible	High
ERI	Level	Interrupt caused by framing or parity (ER).	RIE or REIE	Impossible	↕
RXI	Level	Interrupt caused by receive FIFO data full (RDF).	RIE	Possible	
TXI	Level	Interrupt caused by transmit FIFO data empty (TDFE).	TIE	Possible	
TEI	Level	Interrupt caused by transmit end (TEND).	TEIE	Impossible	
DRI	Level	Interrupt caused by receive data ready (DR).	RIE	Impossible	Low

Note: The TEI and DRI interrupts share the same vector number.
The ERI and BRI interrupts share the same vector number.
If CPU processing is used, clear the flag after the block transfer. If the DMAC is activated, access to the flags is prohibited.

22.6 Serial Port Register (SPTR) and SCIFA-Related Pins

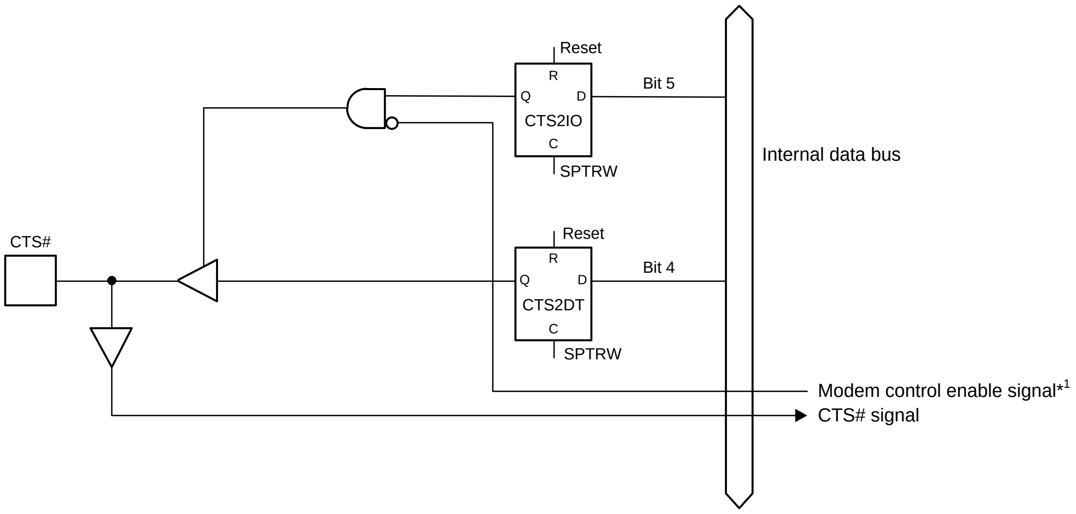
Figure 22.19 to **Figure 22.22** show the relationships between the SPTR register and the SCIFA-related pins.



Note: SPTRW: SPTR write

Note 1. These signals allow the RTS# pin to enter the modem control state with the MCE bits in FCR.

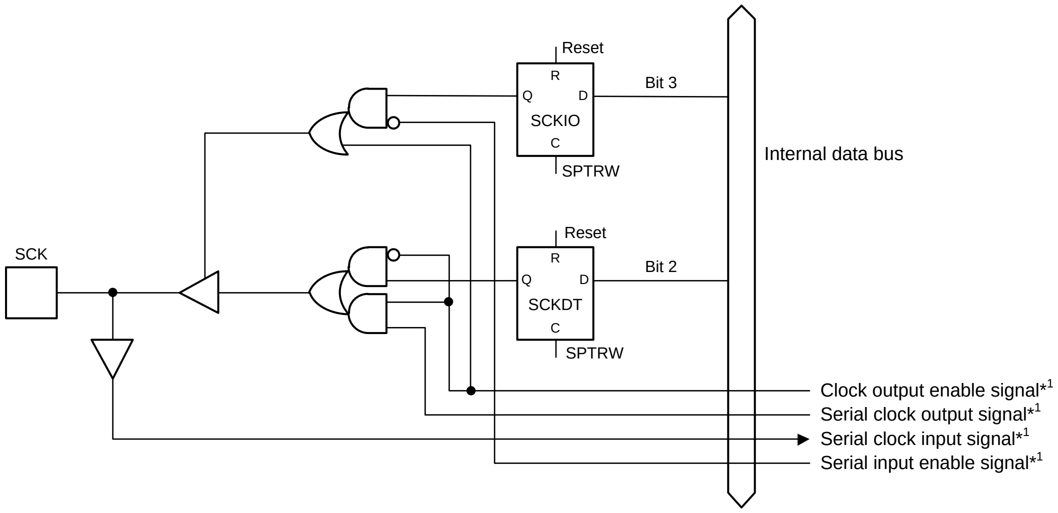
Figure 22.19 RTS2IO Bit and RTS2DT Bit in the SPTR Register, and RTS# Pin



Note: SPTRW: SPTR write

Note 1. These signals allow the CTS# pin to enter the modem control state with the MCE bits in FCR.

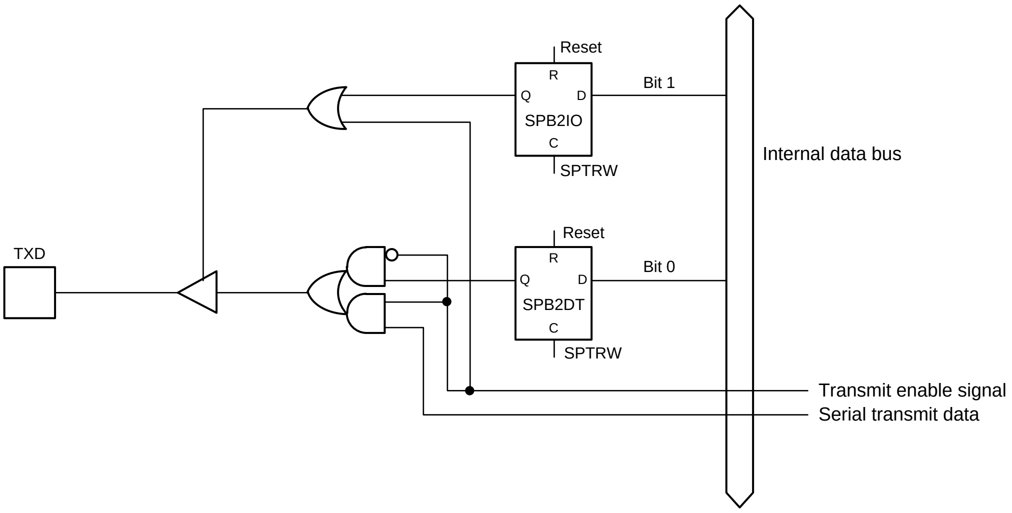
Figure 22.20 CTS2IO Bit and CTS2DT Bit in the SPTR Register, and CTS# Pin



Note: SPTRW: SPTR write

Note 1. These signals control the SCK pin according to the settings of the CM bit in SMR and CKE[1:0] bits in SCR.

Figure 22.21 SCKIO Bit and SCKDT Bit in the SPTR Register, and SCK Pin



Note: SPTRW: SPTR write

Figure 22.22 SPB2IO Bit and SPB2DT Bit in the SPTR Register, and TXD Pin

22.7 Noise Cancellation

Figure 22.23 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of a two-stage flip-flop circuits and a match detection circuit. When the input signals of the noise filter and the output signals of the two-stage flip-flop circuits completely match, the matched level is conveyed as an internal signal. Unless otherwise matched, the previous value is retained. (When the levels sampled on three consecutive cycles of the sampling clock of the noise filter match, the signal is considered valid. If three consecutive sampled values do not match, the signal is considered to be noise rather than a received signal).

In asynchronous mode, the noise cancellation can be applied to the receive signal input to the RXDn pin. The receive level of the RXDn pin is taken in the flip-flop circuit of the noise filter on the base clock (the clock with a frequency 16 or 8 times the transfer rate*[1]).

If the base clock is stopped once with the noise filter enabled and then the base clock input is restarted again, the noise filter operation resumes from the state where the clock was stopped. When SCR.RE is set to 0 during input of the base clock, the noise filter outputs 0 as the internal RxDn signal. The internal match detector continues operating even while operations for reception are stopped, and the result from the last time previous consecutive samples matched is output at the same time as operations for reception are resumed.

Note 1. A frequency 16 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 0, and a frequency 8 times bit rate when either the SEMR.ABCS0 bit or the SEMR.BGDM bit is 1.

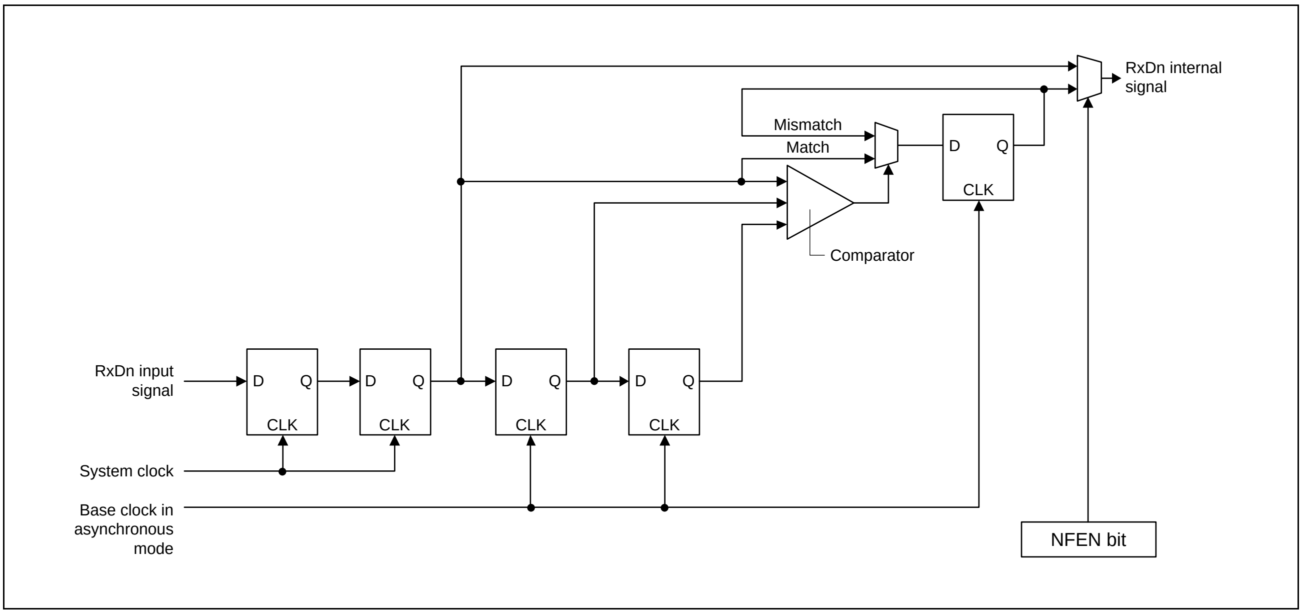


Figure 22.23 Block Diagram of Digital Noise Filter Circuit

22.8 Usage Notes

The following is the notes on using the SCIFA.

22.8.1 FTDR Register Writing and TDFE Flag

The TDFE flag in the serial status register (FSR) is set when the number of transmit data bytes written in the transmit FIFO data register (FTDR) has fallen below the transmission trigger number set by bits TTRG[1:0] in the FIFO control register (FCR) or bits TFTC[4:0] in the FIFO trigger control register (FTCR). After the TDFE flag is set, transmit data up to the number of empty bytes in the FTDR register can be written, allowing efficient continuous transmission.

However, if the number of data bytes written in the FTDR register is equal to or less than the specified transmission trigger number, the TDFE flag will be set to 1 again even after being read as 1 and cleared to 0.

The number of transmit data bytes in the FTDR register can be checked by the 8 higher-order bits of the FIFO data count register (FDR).

22.8.2 FRDR Register Reading and RDF Flag

The RDF flag in the serial status register (FSR) is set when the number of receive data bytes in the receive FIFO data register (FRDR) has become equal to or greater than the reception trigger number set by bits RTRG[1:0] in the FIFO control register (FCR) or bits RFTC[4:0] in the FIFO trigger control register (FTCR). After the RDF flag is set, receive data equivalent to the trigger number can be read from the FRDR register, allowing efficient continuous reception.

However, if the number of data bytes in the FRDR register exceeds the reception trigger number, the RDF flag will be set to 1 again even after being read as 1 and cleared to 0.

The number of receive data bytes in the FRDR register can be checked by the 8 lower-order bits of the FIFO data count register (FDR).

22.8.3 Break Detection and Processing

When a framing error (FER) is detected, a break signal can be detected by reading the RXD pin value directly. In a break, the input from the RXD pin becomes all low. Therefore, the FER flag in the serial status register (FSR) is set to 1 and the parity error flag (PER) may also be set to 1.

Upon detection of a break signal, the SCIFA stops the received data transfer to the FRDR register but continues the receive operation.

22.8.4 Writing to the SPTR Register

b6, b4, b2, and b0 of the SPTR register respectively indicate the input status of their corresponding pins. (See the description of each bit of **Section 22.2.12, Serial Port Register (SPTR)** for details.)

Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

22.8.5 Break Signal Transmission

The output signal from the TXD pin is determined by the SPB2IO bit and the SPB2DT bit in the serial port register (SPTR). The break signal can be sent by using these bits.

The TXD pin does not function as a transmit data output pin during the period from when the SCIFA is initialized to when the TE bit in the SCR register is set to 1 (transmission possible). The TXD pin status during this period is replaced by the SPB2DT bit value. Therefore, the SPB2IO and SPB2DT bits in the SPTR register must have been set to 1 (high output) at first (mark (high) status).

To transmit the break signal during serial transmission, set the SPB2IO bit in the SPTR register to 1, clear the SPB2DT bit to 0 (specify a low level), and then clear the TE bit in the SCR register to 0 (transmission stop). Clearing the TE bit to 0 initializes the transmitter regardless of the current transmission status, and outputs a low level from the TXD pin.

22.8.6 Receive Data Sampling Timing and Receive Margin in Asynchronous Mode

The SCIFA operates on a base clock with a frequency 16 times the transfer rate*[1]. In reception, the SCIFA internally latches the received data at the rising edge of the eighth base clock pulse*[1]. The timing is shown in **Figure 22.24**.

Note 1. This is an example when the SEMR.ABCS0 bit is 0. When the ABCS0 bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock. The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

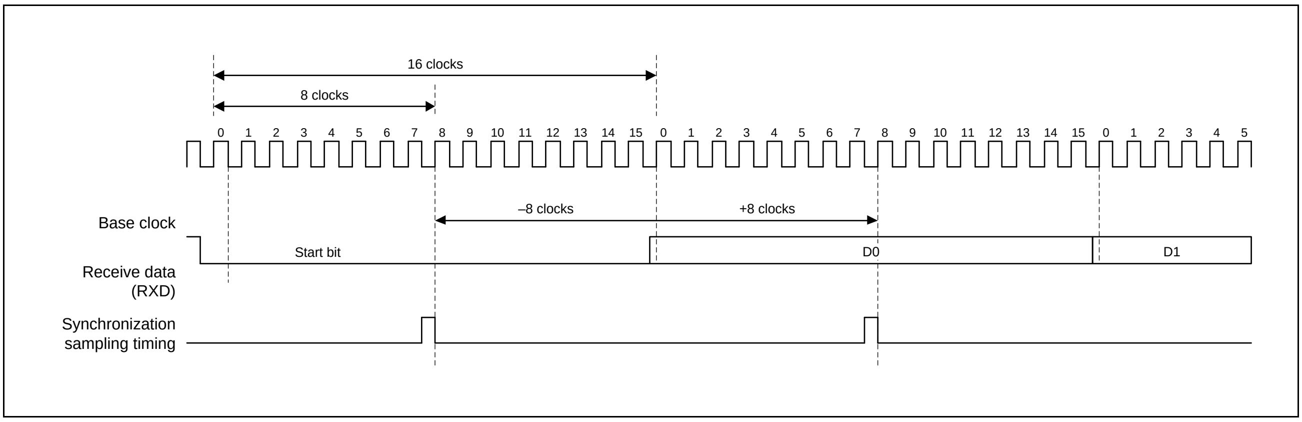


Figure 22.24 Receive Data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

[Equation 1]

$$M = \{0.5 - \frac{1}{2N} - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\} \times 100[\%]$$

Where: M: Receive margin (%)

N: Ratio of clock frequency to bit rate (N = 16)

D: Clock duty (D = 0 to 1.0)

L: Frame length (L = 9 to 12)

F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

[Equation 2]

When D = 0.5 and F = 0:

$$M = (0.5 - 1/(2 \times 16)) \times 100\% = 46.875\%$$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

22.8.7 Note on FER Flag and PER Flag in Serial Status Register (FSR)

The FER flag and PER flag in the serial status register (FSR) are status flags that apply to next entry to be read from the receive FIFO data register (FRDR). After the CPU or DMAC reads the receive FIFO data register, the flags of framing errors and parity errors in the receive data will be cleared. To check the received data for the states of framing errors and parity errors, only read the receive FIFO data register after reading the serial status register.

22.8.8 Notes on External Clock Input in Clock Synchronous Mode

Before setting the TE and RE bits in the serial control register (SCR) to 1, wait for four or more cycles of the peripheral operating clock after the external clock (SCK) is changed from 0 (low) to 1 (high). To input the external clock (SCK) (to start communication), wait for one or more cycles of the external clock after the TE and RE bits in the SCR register are set to 1.

22.8.9 Module Standby Mode Setting

SCIFA operation can be disabled or enabled using the standby control register. As the initial setting, the SCIFA operation is halted. Register access is enabled by clearing module standby mode. For details, refer to **Section 42, Low Power Mode**.

22.8.10 Notes on Operation for Reception when an Internal Clock is Selected in Clock Synchronous Mode

When an internal clock is selected as the clock for reception in clock-synchronous mode, if the number of data stored through the receive FIFO data register (FRDR) becomes equal to or greater than the specified reception trigger number, the RDF flag is set, the RXI interrupt request is generated and, at the same time, output of the synchronizing clock and reception of serial data are stopped. Once the number of data are again less than the specified reception trigger number, output of the synchronizing clock and the reception of serial data are restarted. In addition, if an internal clock is selected for reception in clock synchronous mode, the ORER flag is not set to 1 since no overrun occurs. Accordingly, overruns (indicated by the ORER flag) cannot be used as a BRI interrupt source.

23. Serial Communications Interface (SCIg)

This MCU has two independent serial communications interface (SCI) channels. The SCIs consist of the SCI0 and SCI1 modules, referred to in common as “SCIg modules”.

The SCIg modules (SCI0 and SCI1) can handle both asynchronous and clock synchronous serial communications. Asynchronous serial data communications can be carried out with standard asynchronous communications chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communications Interface Adapter (ACIA). As an extended function in asynchronous communications mode, the SCI also supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards).

23.1 Overview

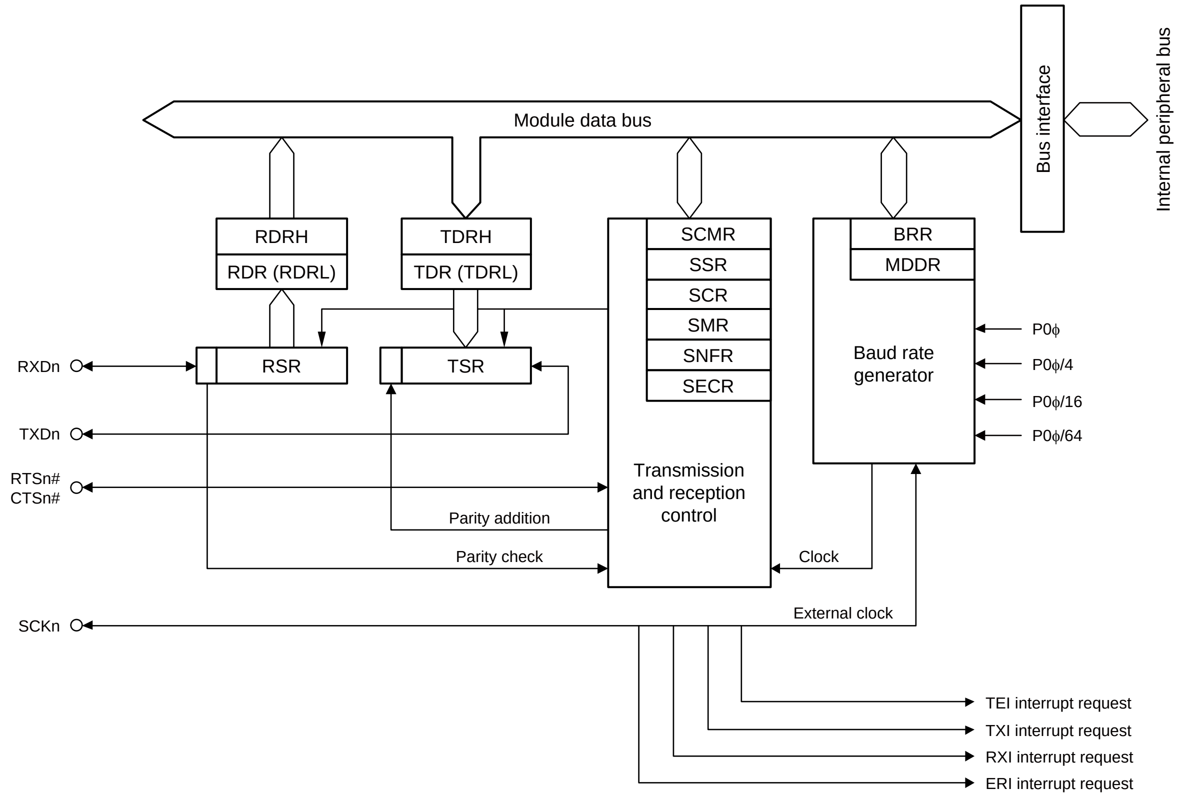
Table 23.1 lists the specifications of the SCIg modules. **Figure 23.1** shows the block diagram of the SCI0 and SCI1 modules.

Table 23.1 SCIg Specifications (1/2)

Item		Description
Serial communications modes		● Asynchronous ● Clock synchronous ● Smart card interface ● IrDA (only for channel 0)
Transfer speed		Bit rate specifiable with the on-chip baud rate generator.
Full-duplex communications		Transmitter: Continuous transmission possible using double-buffer structure. Receiver: Continuous reception possible using double-buffer structure.
I/O pins		See **Table 23.2**.
Data transfer		Selectable as LSB first or MSB first transfer
Interrupt sources		Transmit end, transmit data empty, receive data full, and receive error
Low power consumption function		Module stop state can be set for each channel.
Asynchronous mode	Data length	7, 8, or 9 bits
	Transmission stop bit	1 or 2 bits
	Parity	Even parity, odd parity, or no parity
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.
	Start-bit detection	Low level or falling edge is selectable.
	Break detection	When a framing error occurs, a break can be detected by reading the RXDn pin level directly.
	Clock source	An internal or external clock can be selected.
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Multi-processor communications function	Serial communication among multiple processors
	Noise cancellation	The signal paths from input on the RXDn pins incorporate digital noise filters.
Clock synchronous mode	Data length	8 bits
	Receive error detection	Overrun error
	Hardware flow control	CTSn# and RTSn# pins can be used in controlling transmission/reception.

Table 23.1 SCIg Specifications (2/2)

Item		Description
Smart card interface mode	Error processing	An error signal can be automatically transmitted when detecting a parity error during reception
		Data can be automatically retransmitted when receiving an error signal during transmission
	Data type	Both direct convention and inverse convention are supported.
Bit rate modulation function		Correction of outputs from the on-chip baud rate generator can reduce errors.



RSR: Receive shift register
RDR: Receive data register
TSR: Transmit shift register
TDR: Transmit data register
SMR: Serial mode register
SCR: Serial control register
SSR: Serial status register
SCMR: Smart card mode register
BRR: Bit rate register

MDDR: Modulation duty register
SEMR: Serial extended mode register
SNFR: Noise filter setting register
SECR: Extended function control register
RDRH: Receive data register H
RDRL: Receive data register L
TDRH: Transmit data register H
TDRL: Transmit data register L

Figure 23.1 Block Diagram of SCIg (SCI0 and SCI1)

Table 23.2 lists the pin configuration of the SCIs for the individual modes.

Table 23.2 SCI Pin Configuration in Asynchronous Mode and Clock Synchronous Mode

Channel	Pin Name	I/O	Function
SCI0	SCI0_SCK	I/O	SCI0 clock input/output
	SCI0_RXD	Input	SCI0 receive data input
	SCI0_TXD	Output	SCI0 transmit data output
	SCI0_CTS#/RTS#	I/O	SCI0 transfer start control input/output
SCI1	SCI1_SCK	I/O	SCI1 clock input/output
	SCI1_RXD	Input	SCI1 receive data input
	SCI1_TXD	Output	SCI1 transmit data output
	SCI1_CTS#/RTS#	I/O	SCI1 transfer start control input/output

23.2 Register Descriptions

Table 23.3 Register configuration

BASE Address (Cortex A55 Address Space): (ch0) H’0_1004_D000 (ch1) H’0_1004_D400
BASE Address (Cortex M33 address Space Non Secure): (ch0) H’4004_D000 (ch1) H’4004_D400
BASE Address (Cortex M33 Address Space Secure): (ch0) H’5004_D000 (ch1) H’5004_D400

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Serial Mode Register	SMR	R/W	H’00	BASE(ch0) + H'0000	8
	Bit Rate Register	BRR	R/W	H’FF	BASE(ch0) + H'0001	8
	Serial Control Register	SCR	R/W	H’00	BASE(ch0) + H'0002	8
	Transmit Data Register	TDR	R/W	H’FF	BASE(ch0) + H'0003	8
	Serial Status Register	SSR	R/W*[1]	H’84	BASE(ch0) + H'0004	8
	Receive Data Register	RDR	R	H’00	BASE(ch0) + H'0005	8
	Smart Card Mode Register	SCMR	R/W*[1]	H’F2	BASE(ch0) + H’0006	8
	Serial Extended Mode Register	SEMR	R	H’00	BASE(ch0) + H’0007	8
	Noise Filter Setting Register	SNFR	R	H’00	BASE(ch0) + H’0008	8
	Extended Function Control Register	SECR	R/W	H’00	BASE(ch0) + H’000D	8
	Transmit Data Register H	TDRH	R/W	H’FF	BASE(ch0) + H’000E	8
	Transmit Data Register L	TDRL	R/W	H’FF	BASE(ch0) + H’000F	8
	Transmit Data Register HL	TDRHL	R/W	H’FFFF	BASE(ch0) + H’000E	16
	Receive Data Register H	RDRH	R	H’00	BASE(ch0) + H’0010	8
	Receive Data Register L	RDRL	R	H’00	BASE(ch0) + H’0011	8
	Receive Data Register HL	RDRHL	R	H’0000	BASE(ch0) + H’0010	16
	Modulation Duty Register	MDDR	R/W	H’FF	BASE(ch0) + H’0012	8
1	Serial Mode Register	SMR	R/W	H’00	BASE(ch1) + H’0000	8
	Bit Rate Register	BRR	R/W	H’FF	BASE(ch1) + H’0001	8
	Serial Control Register	SCR	R/W	H’00	BASE(ch1) + H’0002	8
	Transmit Data Register	TDR	R/W	H’FF	BASE(ch1) + H’0003	8
	Serial Status Register	SSR	R/W*[1]	H’84	BASE(ch1) + H’0004	8
	Receive Data Register	RDR	R	H’00	BASE(ch1) + H’0005	8
	Smart Card Mode Register	SCMR	R/W*[1]	H’F2	BASE(ch1) + H’0006	8
	Serial Extended Mode Register	SEMR	R	H’00	BASE(ch1) + H’0007	8
	Noise Filter Setting Register	SNFR	R	H’00	BASE(ch1) + H’0008	8
	Extended Function Control Register	SECR	R/W	H’00	BASE(ch1) + H’000D	8
	Transmit Data Register H	TDRH	R/W	H’FF	BASE(ch1) + H’000E	8
	Transmit Data Register L	TDRL	R/W	H’FF	BASE(ch1) + H’000F	8
	Transmit Data Register HL	TDRHL	R/W	H’FFFF	BASE(ch1) + H’000E	16
	Receive Data Register H	RDRH	R	H’00	BASE(ch1) + H’0010	8
	Receive Data Register L	RDRL	R	H’00	BASE(ch1) + H’0011	8
	Receive Data Register HL	RDRHL	R	H’0000	BASE(ch1) + H’0010	16
	Modulation Duty Register	MDDR	R/W	H’FF	BASE(ch1) + H’0012	8

Note 1. Partly, R only

23.2.1 Receive Shift Register (RSR)

RSR is a shift register which is used to receive serial data input from the RXDn pin and converts it into parallel data. When one frame of data has been received, it is automatically transferred to the RDR register.

The RSR register cannot be directly accessed by the CPU.

23.2.2 Receive Data Register (RDR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

RDR is an 8-bit register that stores receive data.

When one frame of serial data has been received, the received serial data is transferred from RSR to RDR. Then the RSR register can receive the next data.

Since RSR and RDR function as a double buffer in this way, continuous receive operations can be performed.

Read RDR only once after a receive data full interrupt (RXI) has occurred. Note that if next one frame of data is received before reading receive data from RDR, an overrun error occurs.

RDR cannot be written to by the CPU.

23.2.3 Receive Data Register H, L, HL (RDRH, RDRL, RDRHL)

Receive Data Register H (RDRH)

Receive Data Register L (RDRL)

Receive Data Register HL (RDRHL)

	SCIn.RDRHL															
	SCIn.RDRH								SCIn.RDRL							
Bit	b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

RDRH and RDRL are 8-bit registers that store receive data. Use these registers when asynchronous mode and 9-bit data length are selected.

RDRL is the shadow register of RDR; i.e. access to RDRL is equivalent to access to RDR.

After one frame of data is received, the received data is transferred from the RSR register to these registers, thus allowing the RSR register to receive the next data.

The RSR, RDRH and RDRL registers have a double-buffered construction to enable continuous reception.

Read RDRH and RDRL should be performed only once in the order from RDRH to RDRL when a receive data full interrupt (RXI) request is issued. Note that an overrun error occurs when the next frame of data is received before the received data has been read from RDRL.

The CPU cannot write to the RDRH and RDRL registers. Bits 0 to 7 in RDRH are fixed to 0. These bits are read as 0. The write value should be 0.

The RDRHL register can be accessed in 16-bit units.

23.2.4 Transmit Data Register (TDR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TDR is an 8-bit register that stores transmit data.

When the SCI detects that TSR is empty, it transfers the transmit data written in TDR to TSR and starts transmission. The double-buffered structures of TDR and TSR enable continuous serial transmission. If the next transmit data has already been written to TDR when one frame of data is transmitted, the SCI transfers the written data to TSR to continue transmission.

The CPU is able to read from or write to TDR at any time. Only write transmit data to TDR once after each instance of the transmit data empty interrupt (TXI).

23.2.5 Transmit Data Register H, L, HL (TDRH, TDRL, TDRHL)

Transmit Data Register H (TDRH)

Transmit Data Register L (TDRL)

Transmit Data Register HL (TDRHL)

	SCIn.TDRHL															
	SCIn.TDRH								SCIn.TDRL							
Bit	b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TDRH and TDRL are 8-bit registers that store transmit data. Use these registers when asynchronous mode and 9-bit data length are selected.

TDRL is the shadow register of TDR; i.e. access to TDRL is equivalent to access to TDR.

When empty space is detected in the TSR register, the transmit data stored in the TDRH and TDRL registers is transferred to TSR; i.e., transmitting is started.

The TSR, TDRH and TDRL registers have a double-buffered construction to realize continuous reception. When the next data to be transmitted is stored in TDRL after one frame of data has been transmitted, the transmitting operation is continued by transfer to the TSR register.

The CPU can read and write to the TDRH and TDRL registers. Bits 0 to 7 in RDRH are fixed to 1. These bits are read as 1. The write value should be 1.

Writing transmit data to the TDRH and TDRL registers should be performed only once in the order from TDRH to TDRL when a transmit data empty interrupt (TXI) request is issued.

The TDRHL register can be accessed in 16-bit units.

23.2.6 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data.

To perform serial data transmission, the SCI first automatically transfers transmit data from TDR to TSR, and then sends the data to the TXDn pin.

TSR cannot be directly accessed by the CPU.

23.2.7 Serial Mode Register (SMR)

NOTE

Some bits in SMR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4	R/W*4

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKS[1:0]	00b	R/W*4	Clock Select b1 b0 0 0: P0ϕ clock (n = 0)*1 0 1: P0ϕ/4 clock (n = 1)*1 1 0: P0ϕ/16 clock (n = 2)*1 1 1: P0ϕ/64 clock (n = 3)*1
b2	MP	0b	R/W*4	Multi-Processor Mode (Valid only in asynchronous mode) 0: Multi-processor communications function is disabled 1: Multi-processor communications function is enabled
b3	STOP	0b	R/W*4	Stop Bit Length (Valid only in asynchronous mode) 0: 1 stop bit 1: 2 stop bits
b4	PM	0b	R/W*4	Parity Mode (Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity
b5	PE	0b	R/W*4	Parity Enable (Valid only in asynchronous mode) ● When transmitting 0: Parity bit addition is not performed 1: The parity bit is added ● When receiving 0: Parity bit checking is not performed 1: The parity bit is checked
b6	CHR	0b	R/W*4	Character Length (Valid only in asynchronous mode*2) Selects in combination with the SCMR.CHR1 bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*3
b7	CM	0b	R/W*4	Communications Mode 0: Asynchronous mode 1: Clock synchronous mode

Note 1. n is the decimal notation of the value of n in BRR (refer to **Section 23.2.11, Bit Rate Register (BRR)**).

Note 2. In other than asynchronous mode, this bit setting is invalid and a fixed data length of 8 bits is used.

Note 3. LSB first is fixed and the MSB (bit 7) in TDR is not transmitted in transmission.

Note 4. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.

For the relation between the settings of these bits and the baud rate, refer to **Section 23.2.11, Bit Rate Register (BRR)**.

MP Bit (Multi-Processor Mode)

Disables/enables the multi-processor communications function. The settings of the PE bit and PM bit are invalid in multi-processor mode.

STOP Bit (Stop Bit Length)

Selects the stop bit length in transmission.

In reception, only the first stop bit is checked regardless of this bit setting. If the second stop bit is 0, it is treated as the start bit of the next transmit frame.

PM Bit (Parity Mode)

Selects the parity mode (even or odd) for transmission and reception. The setting of the PM bit is invalid in multi-processor mode.

PE Bit (Parity Enable)

When this bit is set to 1, the parity bit is added to transmit data, and the parity bit is checked in reception. Irrespective of the setting of the PE bit, the parity bit is not added or checked in multi-processor format.

CHR Bit (Character Length)

Selects the data length for transmission and reception. Selects in combination with the CHR1 bit in SCMR.

In other than asynchronous mode, a fixed data length of 8 bits is used.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	GM	BLK	PE	PM	BCP[1:0]		CKS[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKS[1:0]	00b	R/W*2	Clock Select b1 b0 0 0: P0ϕ clock (n = 0)*1 0 1: P0ϕ/4 clock (n = 1)*1 1 0: P0ϕ/16 clock (n = 2)*1 1 1: P0ϕ/64 clock (n = 3)*1
b3, b2	BCP[1:0]	00b	R/W*2	Base Clock Pulse Selects the number of base clock cycles in combination with the SCMR.BCP2 bit. **Table 23.4** lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.
b4	PM	0b	R/W*2	Parity Mode (Valid only when the PE bit is 1) 0: Selects even parity 1: Selects odd parity
b5	PE	0b	R/W*2	Parity Enable When this bit is set to 1, a parity bit is added to transmit data, and the parity of received data is checked. Set this bit to 1 in smart card interface mode.
b6	BLK	0b	R/W*2	Block Transfer Mode 0: Normal mode operation 1: Block transfer mode operation
b7	GM	0b	R/W*2	GSM Mode 0: Normal mode operation 1: GSM mode operation

Note 1. n is the decimal notation of the value of n in BRR (refer to **Section 23.2.11, Bit Rate Register (BRR)**).

Note 2. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.

For the relationship between the settings of these bits and the baud rate, refer to **Section 23.2.11, Bit Rate Register (BRR)**.

BCP[1:0] Bits (Base Clock Pulse)

These bits select the number of base clock cycles in a 1-bit data transfer time in smart card interface mode. Set these bits in combination with the SCMR.BCP2 bit.

For details, refer to **Section 23.6.4, Receive Data Sampling Timing and Reception Margin**.

Table 23.4 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)*[1]
0	0	1	128 clock cycles (S = 128)*[1]
0	1	0	186 clock cycles (S = 186)*[1]
0	1	1	512 clock cycles (S = 512)*[1]
1	0	0	32 clock cycles (S = 32)*[1] (Initial Value)
1	0	1	64 clock cycles (S = 64)*[1]
1	1	0	372 clock cycles (S = 372)*[1]
1	1	1	256 clock cycles (S = 256)*[1]

Note 1. S is the value of Base Clock Cycles for 1-bit Period in BRR (refer to **Section 23.2.11, Bit Rate Register (BRR)**, **Table 23.8 Base Clock Settings in Smart Card Interface Mode**).

PM Bit (Parity Mode)

Selects the parity mode for transmission and reception (even or odd).

For details on the usage of this bit in smart card interface mode, refer to **Section 23.6.2, Data Format (Except in Block Transfer Mode)**.

PE Bit (Parity Enable)

Set the PE bit to 1.

The parity bit is added to transmit data before transmission, and the parity bit is checked in reception.

BLK Bit (Block Transfer Mode)

Setting this bit to 1 allows block transfer mode operation.

For details, refer to **Section 23.6.3, Block Transfer Mode**.

GM Bit (GSM Mode)

Setting this bit to 1 allows GSM mode operation.

In GSM mode, the SSR.TEND flag set timing is put forward to 11.0 etu (elementary time unit = 1-bit transfer time) from the start and the clock output control function is appended. For details, refer to **Section 23.6.6, Serial Data Transmission (Except in Block Transfer Mode)** and **Section 23.6.8, Clock Output Control**.

23.2.8 Serial Control Register (SCR)

NOTE

Some bits in SCR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W*2	R/W*2	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKE[1:0]	00b	R/W*1	Clock Enable ● For SCI0 and SCI1 (Asynchronous mode) b1 b0 0 0: On-chip baud rate generator The SCKn pin functions as I/O port. 0 1: On-chip baud rate generator The clock with the same frequency as the bit rate is output from the SCKn pin. 1 x: External clock The clock with a frequency 16 times the bit rate should be input from the SCKn pin. Input a clock signal with a frequency eight times the bit rate when the SEMR.ABCS bit is 1. (Clock synchronous mode) b1 b0 0 x: Internal clock The SCKn pin functions as the clock output pin. 1 x: External clock The SCKn pin functions as the clock input pin.
b2	TEIE	0b	R/W	Transmit End Interrupt Enable 0: A TEI interrupt request is disabled 1: A TEI interrupt request is enabled
b3	MPIE	0b	R/W	Multi-Processor Interrupt Enable (Valid in asynchronous mode when SMR.MP = 1) 0: Normal reception 1: When the data with the multi-processor bit set to 0 is received, the data is not read, and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE bit is automatically cleared to 0, and normal reception is resumed.
b4	RE	0b	R/W*2	Receive Enable 0: Serial reception is disabled 1: Serial reception is enabled
b5	TE	0b	R/W*2	Transmit Enable 0: Serial transmission is disabled 1: Serial transmission is enabled
b6	RIE	0b	R/W	Receive Interrupt Enable 0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled

Bit	Bit Name	Initial Value	R/W	Description
b7	TIE	0b	R/W	Transmit Interrupt Enable 0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled

Remarks: x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. 1 can be written only when TE = 0 and RE = 0, while the SMR.CM bit is 1. After setting TE or RE to 1, only 0 can be written to TE and RE. While the SMR.CM bit is 0 and the SIMR1.IICM bit is 0, writing is enabled under any condition.

CKE[1:0] Bits (Clock Enable)

These bits select the clock source and SCKn pin function.

The combination of the settings of these bits and of the SEMR.ACS0 bit sets the internal TMR clock.

TEIE Bit (Transmit End Interrupt Enable)

Enables or disables a TEI interrupt request.

A TEI interrupt request is disabled by setting the TEIE bit to 0.

MPIE Bit (Multi-Processor Interrupt Enable)

When this bit is set to 1 and the data with the multi-processor bit set to 0 is received, the data is not read and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE is automatically cleared to 0, and normal reception is resumed. For details, refer to **Section 23.4, Multi-Processor Communications Function**.

When the receive data includes the MPB bit is SSR set to 0, the receive data is not transferred from the RSR to the RDR, a receive error is not detected, and setting the flags ORER and FER to 1 is disabled.

When the receive data includes the MPB bit set to 1, the MPB bit is set to 1, the MPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the RIE bit in SCR is set to 1), and setting the flags ORER and FER to 1 is enabled.

MPIE should be set to 0 if multi-processor communications function is not to be used.

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit in asynchronous mode or the synchronous clock input in clock synchronous mode. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by setting the RIE bit to 0.

An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in SSR and then setting the flag to 0, or setting the RIE bit to 0.

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W*2	R/W*2	R/W	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	CKE[1:0]	00b	R/W*1	Clock Enable ● When SMR.GM = 0 b1 b0 0 0: Output disabled (The SCKn pin is available for use as an I/O port according to the I/O port settings.) 0 1: Clock output 1 x: (Setting prohibited) ● When SMR.GM = 1 b1 b0 0 0: Output fixed low x 1: Clock output 1 0: Output fixed high
b2	TEIE	0b	R/W	Transmit End Interrupt Enable This bit should be 0 in smart card interface mode.
b3	MPIE	0b	R/W	Multi-Processor Interrupt Enable This bit should be 0 in smart card interface mode.
b4	RE	0b	R/W*2	Receive Enable 0: Serial reception is disabled 1: Serial reception is enabled
b5	TE	0b	R/W*2	Transmit Enable 0: Serial transmission is disabled 1: Serial transmission is enabled
b6	RIE	0b	R/W	Receive Interrupt Enable 0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled
b7	TIE	0b	R/W	Transmit Interrupt Enable 0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled

Remarks: x: Don’t care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. 1 can be written only when TE = 0 and RE = 0. After setting TE or RE to 1, only 0 can be written in TE and RE.

For details on interrupt requests, refer to **Section 23.8, Interrupt Sources**.

CKE[1:0] Bits (Clock Enable)

These bits control the clock output from the SCKn pin.

In GSM mode, clock output can be dynamically switched. For details, refer to **Section 23.6.8, Clock Output Control**.

TEIE Bit (Transmit End Interrupt Enable)

This bit should be 0 in smart card interface mode.

MPIE Bit (Multi-Processor Interrupt Enable)

This bit should be 0 in smart card interface mode.

RE Bit (Receive Enable)

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by setting the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

TE Bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

RIE Bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by setting the RIE bit to 0.

An ERI interrupt request can be canceled by reading 1 from the ORER, FER, or PER flag in SSR and then setting the flag to 0, or setting the RIE bit to 0.

23.2.9 Serial Status Register (SSR)

Some bits in SSR have different functions in smart card interface mode and non-smart card interface mode.

(1) Non-Smart Card Interface Mode (SCMR.SMIF = 0)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Initial Value	1	0	0	0	0	1	0	0
R/W	R/(W)^{*2}	R/(W)^{*2}	R/(W)^{*1}	R/(W)^{*1}	R/(W)^{*1}	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	MPBT	0b	R/W	Multi-Processor Bit Transfer Sets the multi-processor bit for adding to the transmission frame 0: Data transmission cycles 1: ID transmission cycles
b1	MPB	0b	R	Multi-Processor Value of the multi-processor bit in the reception frame 0: Data transmission cycles 1: ID transmission cycles
b2	TEND	1b	R	Transmit End Flag 0: A character is being transmitted. 1: Character transfer has been completed.
b3	PER	0b	R/(W)^{*1}	Parity Error Flag 0: No parity error occurred 1: A parity error has occurred
b4	FER	0b	R/(W)^{*1}	Framing Error Flag 0: No framing error occurred 1: A framing error has occurred
b5	ORER	0b	R/(W)^{*1}	Overrun Error Flag 0: No overrun error occurred 1: An overrun error has occurred
b6	RDRF	0b	R/(W)^{*2}	Receive Data Full Flag 0: When data is transferred from RDR 1: When data has been received normally, and transferred from RSR to RDR
b7	TDRE	1b	R/(W)^{*2}	Transmit Data Empty Flag 0: When data is transferred to TDR 1: When data is transferred from TDR to TSR

Note 1. Only 0 can be written to this bit, to clear the flag.

Note 2. Write 1 when writing is necessary.

MPB Bit (Multi-Processor)

Holds the value of the multi-processor bit in the reception frame. This bit does not change when the SCR.RE bit is 0.

TEND Flag (Transmit End Flag)

Indicates completion of transmission.

[Setting conditions]

- When the SCR.TE bit is set to 0 (serial transmission is disabled)
  When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When the TDR register is not updated at the time of transmission of the tail-end bit of a character being transmitted

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception
  Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1
  Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

FER Flag (Framing Error Flag)

Indicates that a framing error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When the stop bit is 0
  In 2-stop-bit mode, only the first stop bit is checked whether it is 1 but the second stop bit is not checked. Note that although receive data when the framing error occurs is transferred to RDR, no RXI interrupt request occurs. In addition, when the FER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to FER after reading FER = 1
  Even when the SCR.RE bit is set to 0, the FER flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from RDR
  In RDR, receive data prior to an overrun error occurrence is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed. Note that, in clock synchronous mode, serial transmission also cannot continue.

[Clearing condition]

- When 0 is written to ORER after reading ORER = 1
  Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether RDR has received data.

[Setting condition]

- When data has been received normally, and transferred from RSR to RDR

[Clearing condition]

- When data is transferred from RDR

TDRE Flag (Transmit Data Empty Flag)

Indicates whether TDR has data to be transmitted.

[Setting condition]

- When data is transferred from TDR to TSR

[Clearing condition]

- When data is transferred to TDR

(2) Smart Card Interface Mode (SCMR.SMIF = 1)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	RDRF	ORER	ERS	PER	TEND	MPB	MPBT
Initial Value	1	0	0	0	0	1	0	0
R/W	R/(W)*2	R/(W)*2	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	MPBT	0b	R/W	Multi-Processor Bit Transfer This bit should be set to 0 in smart card interface mode.
b1	MPB	0b	R	Multi-Processor This bit is not used in smart card interface mode. It should be set to 0.
b2	TEND	1b	R	Transmit End Flag 0: A character is being transmitted. 1: Character transfer has been completed.
b3	PER	0b	R/(W) *1	Parity Error Flag 0: No parity error occurred 1: A parity error has occurred
b4	ERS	0b	R/(W) *1	Error Signal Status Flag 0: Low error signal not responded 1: Low error signal responded
b5	ORER	0b	R/(W) *1	Overrun Error Flag 0: No overrun error occurred 1: An overrun error has occurred
b6	RDRF	0b	R/(W) *2	Receive Data Full Flag 0: When data is transferred from RDR 1: When data has been received normally, and transferred from RSR to RDR
b7	TDRE	1b	R/(W) *2	Transmit Data Empty Flag 0: When data is transferred to TDR 1: When data is transferred from TDR to TSR

Note 1. Only 0 can be written to this bit, to clear the flag.

Note 2. Write 1 when writing is necessary.

TEND Flag (Transmit End Flag)

With no error signal from the receiving side, this bit is set to 1 when further data for transfer is ready to be transferred to the TDR register.

[Setting conditions]

- When the SCR.TE bit = 0 (serial transmission is disabled)
  When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When a specified period has elapsed after the latest transmission of 1 byte, the ERS flag is 0, and the TDR register is not updated
  The set timing is determined by register settings as listed below.
  When SMR.GM = 0 and SMR.BLK = 0, 12.5 etu after the start of transmission
  When SMR.GM = 0 and SMR.BLK = 1, 11.5 etu after the start of transmission
  When SMR.GM = 1 and SMR.BLK = 0, 11.0 etu after the start of transmission
  When SMR.GM = 1 and SMR.BLK = 1, 11.0 etu after the start of transmission

[Clearing condition]

- When transmit data are written to the TDR register while the SCR.TE bit is 1

PER Flag (Parity Error Flag)

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception
  Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1
  Even when the SCR.RE bit is set to 0 (serial reception is disabled), the PER flag is not affected and retains its previous value.

ERS Flag (Error Signal Status Flag)

[Setting condition]

- When a low error signal is sampled

[Clearing condition]

- When 0 is written to ERS after reading ERS = 1
  Even when the SCR.RE bit is set to 0, the ERS flag is not affected and retains its previous value.

ORER Flag (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from RDR
  In RDR, the receive data prior to an overrun error occurrence is retained, but data received following the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed.

[Clearing condition]

- When 0 is written to ORER after reading ORER = 1
  Even when the SCR.RE bit is set to 0, the ORER flag is not affected and retains its previous value.

RDRF Flag (Receive Data Full Flag)

Indicates whether RDR has received data.

[Setting condition]

- When data has been received normally, and transferred from RSR to RDR

[Clearing condition]

- When data is transferred from RDR

TDRE Flag (Transmit Data Empty Flag)

Indicates whether TDR has data to be transmitted.

[Setting condition]

- When data is transferred from TDR to TSR

[Clearing condition]

- When data is transferred to TDR

23.2.10 Smart Card Mode Register (SCMR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	BCP2	—	—	CHR1	SDIR	SINV	—	SMIF
Initial Value	1	1	1	1	0	0	1	0
R/W	R/W*1	R/W	R/W	R/W*1	R/W*1	R/W*1	R/W	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	SMIF	0b	R/W*1	Smart Card Interface Mode Select 0: Non-smart card interface mode (Asynchronous mode, clock synchronous mode,) 1: Smart card interface mode
b1	—	1b	R/W	Reserved This bit is read as 1. The write value should be 1.
b2	SINV	0b	R/W*1	Transmitted/Received Data Invert 0: TDR contents are transmitted as they are. Receive data is stored as it is in RDR. 1: TDR contents are inverted before being transmitted. Receive data is stored in inverted form in RDR.
b3	SDIR	0b	R/W*1	Transmitted/Received Data Transfer Direction This bit can be used in the following modes. ● Smart card interface mode ● Asynchronous mode (multi-processor mode) ● Clock synchronous mode 0: Transfer with LSB first 1: Transfer with MSB first
b4	CHR1	1b	R/W*1	Character Length 1 (Only valid in asynchronous mode)*2 Selects in combination with the SMR.CHR bit. CHR1 CHR 0 0: Transmit/receive in 9-bit data length 0 1: Transmit/receive in 9-bit data length 1 0: Transmit/receive in 8-bit data length (initial value) 1 1: Transmit/receive in 7-bit data length*3
b6, b5	—	1b	R/W	Reserved These bits are read as 1. The write value should be 1.
b7	BCP2	1b	R/W*1	Base Clock Pulse 2 Selects the number of base clock cycles in combination with the SMR.BCP[1:0] bits. **Table 23.5** lists the combinations of the SCMR.BCP2 bit and SMR.BCP[1:0] bits.

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

Note 2. The setting is invalid and a fixed data length of 8 bits is used in modes other than asynchronous mode.

Note 3. LSB first should be selected and the value of MSB (b7) in TDR cannot be transmitted.

SMIF Bit (Smart Card Interface Mode Select)

When this bit is set to 1, smart card interface mode is selected.

When this bit is set to 0, non-smart card interface mode, i.e., asynchronous mode (including multi-processor mode), clock synchronous mode is selected.

SINV Bit (Transmitted/Received Data Invert)

Inverts the transmit/receive data logic level. This bit does not affect the logic level of the parity bit. To invert the parity bit, invert the PM bit in SMR.

CHR1 bit (Character Length 1)

Selects the data length of transmit/receive data. Selects in combination with the CHR bit in SMR.

A fixed data length of 8 bits is used in modes other than asynchronous mode.

BCP2 Bit (Base Clock Pulse 2)

Selects the number of base clock cycles in a 1-bit data transfer time in smart card interface mode. Set this bit in combination with the SMR.BCP[1:0] bits.

Table 23.5 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits

SCMR.BCP2 Bit	SMR.BCP[1:0] Bits		Number of Base Clock Cycles for 1-Bit Transfer Period
0	0	0	93 clock cycles (S = 93)*1
0	0	1	128 clock cycles (S = 128)*1
0	1	0	186 clock cycles (S = 186)*1
0	1	1	512 clock cycles (S = 512)*1
1	0	0	32 clock cycles (S = 32)*1 (Initial Value)
1	0	1	64 clock cycles (S = 64)*1
1	1	0	372 clock cycles (S = 372)*1
1	1	1	256 clock cycles (S = 256)*1

Note 1. S is the value of Base Clock Cycles for 1-bit Period BRR (refer to **Section 23.2.11, Bit Rate Register (BRR)**, **Table 23.8 Base Clock Settings in Smart Card Interface Mode**).

23.2.11 Bit Rate Register (BRR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

BRR is an 8-bit register that adjusts the bit rate.

As each SCI channel has independent baud rate generator control, different bit rates can be set for each. **Table 23.6** shows the relationship between the setting (N) in the BRR and the bit rate (B) for normal asynchronous mode, multi-processor communication, clock synchronous mode, and smart card interface mode.

The initial value of BRR is H'FF.

BRR can be read from by the CPU, but it can be written to only when the TE and RE bits in SCR are 0.

Table 23.6 Relationship between N Setting in BRR and Bit Rate B

Mode	SEMR Setting		BRR Setting	Error
	BGDM Bit	ABCS Bit		
Asynchronous or multi-processor communication	0	0	$N = \frac{P0\phi \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	$\text{Error}\ (\%) = \{\frac{P0\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1\} \times 100$
	0	1	$N = \frac{P0\phi \times 10^6}{32 \times 2^{2n-1} \times B} - 1$	$\text{Error}\ (\%) = \{\frac{P0\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1\} \times 100$
	1	0		
	1	1	$N = \frac{P0\phi \times 10^6}{16 \times 2^{2n-1} \times B} - 1$	$\text{Error}\ (\%) = \{\frac{P0\phi \times 10^6}{B \times 16 \times 2^{2n-1} \times (N+1)} - 1\} \times 100$
Clock synchronous			$N = \frac{P0\phi \times 10^6}{8 \times 2^{2n-1} \times B} - 1$	
Smart card interface			$N = \frac{P0\phi \times 10^6}{S \times 2^{2n-1} \times B} - 1$	$\text{Error}\ (\%) = \{\frac{P0\phi \times 10^6}{B \times S \times 2^{2n-1} \times (N+1)} - 1\} \times 100$

Remarks: B: Bit rate (bps)
N: BRR setting for on-chip baud rate generator (0 ≤ N ≤ 255)
P0ϕ: Operating frequency (MHz)
n and S: Determined by the settings of the SMR and SCMR registers as listed in the table below.

Table 23.7 Clock Source Settings

SMR.CKS[1:0] Bit Setting	Clock Source	n
0 0	P0ϕ clock	0
0 1	P0ϕ/4 clock	1
1 0	P0ϕ/16 clock	2
1 1	P0ϕ/64 clock	3

Table 23.8 Base Clock Settings in Smart Card Interface Mode

SCMR.BCP2 Bit Setting	SMR.BCP[1:0] Bit Setting	Base Clock Cycles for 1-bit Period	S
0	0 0	93 clock cycles	93
0	0 1	128 clock cycles	128
0	1 0	186 clock cycles	186
0	1 1	512 clock cycles	512
1	0 0	32 clock cycles	32
1	0 1	64 clock cycles	64
1	1 0	372 clock cycles	372
1	1 1	256 clock cycles	256

Table 23.9 lists examples of N settings in BRR in normal asynchronous mode. **Table 23.10** lists the maximum bit rate settable for each operating frequency. Examples of BRR (N) settings in clock synchronous mode are listed in **Table 23.12**. Examples of BRR (N) settings in smart card interface mode are listed in **Table 23.14**. In smart card interface mode, the number of base clock cycles S in a 1-bit data transfer time can be selected. For details, refer to **Section 23.6.4, Receive Data Sampling Timing and Reception Margin**. **Table 23.11** and **Table 23.13** list the maximum bit rates with external clock input.

When either the asynchronous mode base clock select bit (ABCS) or the baud rate generator double-speed mode select bit (BGDM) in the serial extended mode register (SEMR) is set to 1 in asynchronous mode, the bit rate becomes twice that listed in **Table 23.9**. When both of those registers are set to 1, the bit rate becomes four times the listed value.

Table 23.9 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode)

	Operating Frequency P0ϕ (MHz)		
	100		
Bit Rate (bps)	n	N	Error (%)
110	—	—	—
150	—	—	—
300	3	162	−0.15
600	3	80	0.47
1200	2	162	−0.15
2400	2	80	0.47
4800	1	162	−0.15
9600	1	80	0.47
19200	0	162	−0.15
31250	0	99	0.00
38400	0	80	0.47

Note: This is an example when the ABCS and BGDM bits in SEMR are 0.
When either the ABCS bit or BGDM bit is set to 1, the bit rate doubles.
When both ABCS and BGDM bits in SEMR are set to 1, the bit rate increases four times.

Table 23.10 Maximum Bit Rate for Each Operating Frequency (Asynchronous Mode)

	SEMR Settings				
P0ϕ (MHz)	BGDM Bit	ABCS Bit	n	N	Maximum Bit Rate (bps)
100	0	0	0	0	3125000
		1	0	0	6250000
	1	0	0	0	6250000
		1	0	0	12500000

Table 23.11 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

	Maximum Bit Rate (bps)		
P0ϕ (MHz)	External Input Clock (MHz)	SEMR.ABCS Bit = 0	SEMR.ABCS Bit = 1
100	25.000	1562500	3125000

Table 23.12 BRR Settings for Various Bit Rates (Clock Synchronous Mode)

	Operating Frequency P0ϕ (MHz)	
	100	
Bit Rate (bps)	n	N
110	—	—
250	—	—
500	—	—
1 k	—	—
2.5 k	3	155
5 k	3	77
10 k	2	155
25 k	1	249
50 k	1	124
100 k	0	249
250 k	0	99
500 k	0	49
1 M	0	24
2.5 M	0	9
5 M	0	4
7.5 M	0	2

Note: Continuous transmission or reception is impossible. After transmitting or receiving one frame of data, there is an interval of a 1-bit period before starting transmitting or receiving the next frame of data. The output of the synchronization clock is stopped for a 1-bit period. For this reason, it takes 9 bits worth of time to transfer one frame (8 bits) of data, and the average transfer rate is 8/9 times the bit rate.

Remarks: Space: Setting prohibited.
—: Can be set, but an error will occur.

Table 23.13 Maximum Bit Rate with External Clock Input (Clock Synchronous Mode)

P0ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (Mbps)
100	16.6667	16.6667

Table 23.14 BRR Settings for Various Bit Rates (Smart Card Interface Mode, n = 0, S = 372)

Bit Rate (bps)	P0ϕ (MHz)	n	N	Error (%)
9600	100	0	13	0.01

Table 23.15 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 32)

P0ϕ (MHz)	Maximum Bit Rate (bps)	n	N
100.00	1562500	0	0

Table 23.16 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 372)

P0ϕ (MHz)	Maximum Bit Rate (bps)	n	N
100.00	134409	0	0

23.2.12 Modulation Duty Register (MDDR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MDDR corrects the bit rate adjusted by the BRR register.

When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (M/256). The relationship between the MDDR setting (M) and the bit rate (B) is given in **Table 23.17**.

The initial value of MDDR is H'FF. Bit 7 in this register is fixed to 1.

The CPU can read the MDDR register, but this register is only writable when the TE and RE bits in SCR are 0.

Table 23.17 Relationship between MDDR Setting (M) and Bit Rate (B) When Bit Rate Modulation Function is Used

	SEMR	Setting		
Mode	BGDM Bit	ABCS Bit	BRR Setting	Error
Asynchronous or multi-processor communication	0	0	$N=\frac{P0\emptyset\times10^{6}}{64\times2^{2n-1}\times(256/M)\times B}-1$	$\text{Error}\,(\%)=\{\frac{P0\emptyset\times10^{6}}{B\times64\times2^{2n-1}\times(256M)\times(N+1)}-1\}\times100$
	0	1	$N=\frac{P0\emptyset\times10^{6}}{32\times2^{2n-1}\times(256/M)\times B}-1$	$\text{Error}\,(\%)=\{\frac{P0\emptyset\times10^{6}}{B\times32\times2^{2n-1}\times(256M)\times(N+1)}-1\}\times100$
	1	0		
	1	1	$N=\frac{P0\emptyset\times10^{6}}{16\times2^{2n-1}\times(256/M)\times B}-1$	$\text{Error}\,(\%)=\{\frac{P0\emptyset\times10^{6}}{B\times16\times2^{2n-1}\times(256M)\times(N+1)}-1\}\times100$
Clock synchronous			$N=\frac{P0\emptyset\times10^{6}}{8\times2^{2n-1}\times(256/M)\times B}-1$	
Smart card interface			$N=\frac{P0\emptyset\times10^{6}}{S\times2^{2n-1}\times(256/M)\times B}-1$	$\text{Error}\,(\%)=\{\frac{P0\emptyset\times10^{6}}{B\times S\times2^{2n-1}\times(256M)\times(N+1)}-1\}\times100$

Note: Do not use this function in clock synchronous mode with the highest speed setting (SMR.CKS[1:0] = 00b, SCR.CKE[1] = 0, and BRR = 0).

Remarks: B: Bit rate (bps)
M: MDDR setting (128 ≤ MDDR ≤ 256)
N: BRR setting for baud rate generator (0 ≤ N ≤ 255)
P0ϕ: Operating frequency (MHz)
n and S: Determined by the settings of the SMR and SCMR registers as listed in **Table 23.7** and **Table 23.8**, **Section 23.2.11, Bit Rate Register (BRR)**.

23.2.13 Serial Extended Mode Register (SEMR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	RXDESEL	BGDM	NFEN	ABCS	—	BRME	—	ACS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W	R/W*1	R/W*1	R/W	R/W	R/W	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
b0	ACS0	0b	R/W*1	Asynchronous Mode Clock Source Select (Valid only in asynchronous mode) 0: External clock input 1: Reserved bit
b1	—	0b	R/W	Reserved This bit is read as 0. The write value should be 0.
b2	BRME	0b	R/W	Bit Rate Modulation Enable 0: Bit rate modulation function is disabled. 1: Bit rate modulation function is enabled.
b3	—	0b	R/W	Reserved This bit is read as 0. The write value should be 0.
b4	ABCS	0b	R/W*1	Asynchronous Mode Base Clock Select (Valid only in asynchronous mode) 0: Selects 16 base clock cycles for 1-bit period. 1: Selects 8 base clock cycles for 1-bit period.
b5	NFEN	0b	R/W*1	Digital Noise Filter Function Enable (In asynchronous mode) 0: Noise cancellation function for the RXDn input signal is disabled. 1: Noise cancellation function for the RXDn input signal is enabled.
b6	BGDM	0b	R/W	Baud Rate Generator Double-Speed Mode Select (Only valid the CKE[1] bit in SCR is 0 in asynchronous mode). 0: Baud rate generator outputs the clock with normal frequency. 1: Baud rate generator outputs the clock with doubled frequency.
b7	RXDESEL	0b	R/W*1	Asynchronous Start Bit Edge Detection Select (Valid only in asynchronous mode) 0: The low level on the RXDn pin is detected as the start bit. 1: A falling edge on the RXDn pin is detected as the start bit.

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

SEMR selects the clock source for 1-bit period in asynchronous mode.

ACS0 Bit (Asynchronous Mode Clock Source Select)

Selects the clock source in the asynchronous mode.

The ACS0 bit is valid in asynchronous mode (CM bit in SMR = 0) and when an external clock input is selected (CKE[1:0] bits in SCR = 10b or 11b).

Set the ACS0 bit to 0 in other than asynchronous mode.

BRME bit (Bit Rate Modulation Enable)

Enables and disables the bit rate modulation function. The bit rate generated by on-chip baud rate generator is evenly corrected when this function is enabled.

NFEN Bit (Digital Noise Filter Function Enable)

This bit enables or disables the digital noise filter function.

When the function is enabled, noise cancellation is applied to the RXDn input signal in asynchronous mode. In any mode other than above, set the NFEN bit to 0 to disable the digital noise filter function.

When the function is disabled, input signals are transferred as is, as internal signals.

BGDM bit (Baud Rate Generator Double-Speed Mode Select)

Selects the cycle of output clock for the baud rate generator.

This bit is valid when the on-chip baud rate generator is selected as the clock source (SCR.CKE[1] = 0) in asynchronous mode (SMR.CM = 0). For the clock output from the baud rate generator, either normal or doubled frequency can be selected. The base clock is generated by the clock output from the baud rate generator. When the BGDM bit is set to 1, the base clock cycle is halved and the bit rate is doubled. Set this bit to 0 in modes other than asynchronous mode.

RXDESEL Bit (Asynchronous Start Bit Edge Detection Select)

Selects the detection method of the start bit for reception in asynchronous mode. When a break occurs, data receiving operation depends on the settings of this bit. Set this bit to 1 when reception should be stopped while a break occurs or when reception should be started without retaining the RXDn pin input at high level for the period of one data frame or longer after completion of the break.

Set this bit to 0 in modes other than asynchronous mode.

23.2.14 Noise Filter Setting Register (SNFR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	NFCS[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W*[1]	R/W*[1]	R/W*[1]

Bit	Bit Name	Initial Value	R/W	Description
b2 to b0	NFCS[2:0]	All 0	R/W*[1]	Noise Filter Clock Select In asynchronous mode, the standard setting for the base clock is as follows. b2 b0 0 0 0: The clock signal divided by 1 is used with the noise filter. Settings other than above are prohibited.
b7 to b3	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR are 0 (serial reception and transmission disabled).

NFCS[2:0] Bits (Noise Filter Clock Select)

These bits select the sampling clock for the digital noise filter. To use the noise filter in asynchronous mode, set these bits to 000b.

23.2.15 Extended function control register (SECR)

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	CTSE	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W*1	R/W

Bit	Bit Name	Initial Value	R/W	Description
b0	—	0b	R/W	Reserved This bit is read as 0. The write value should be 0.
b1	CTSE	0b	R/W*1	CTS Enable 0: CTS function is disabled (RTS output function is enabled). 1: CTS function is enabled.
B7 to b2	—	All 0	R/W	Reserved These bits are read as 0. The write value should be 0.

Note 1. Writing to the bit is only possible when the SCR.RE and SCR.TE bits are 0.

SECR is used to select the extension settings in asynchronous and clock synchronous modes.

CTSE Bit (CTS Enable)

Set this bit to 1 if the SSn# pin is to be used for inputting of the CTS control signal to control of transmission and reception. The RTS signal is output when this bit is set to 0. Set this bit to 0 in smart card interface mode. Do not set both the CTSE and SSE bits to enabled (even if this setting is made, operation is the same as that when these bits are set to 0).

23.3 Operation in Asynchronous Mode

Figure 23.2 shows the general format for asynchronous serial communications.

One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level).

In asynchronous serial communications, the communications line is usually held in the mark state (high level).

The SCI monitors the communications line. When the SCI detects a low, it regards that as a start bit and starts serial communication.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

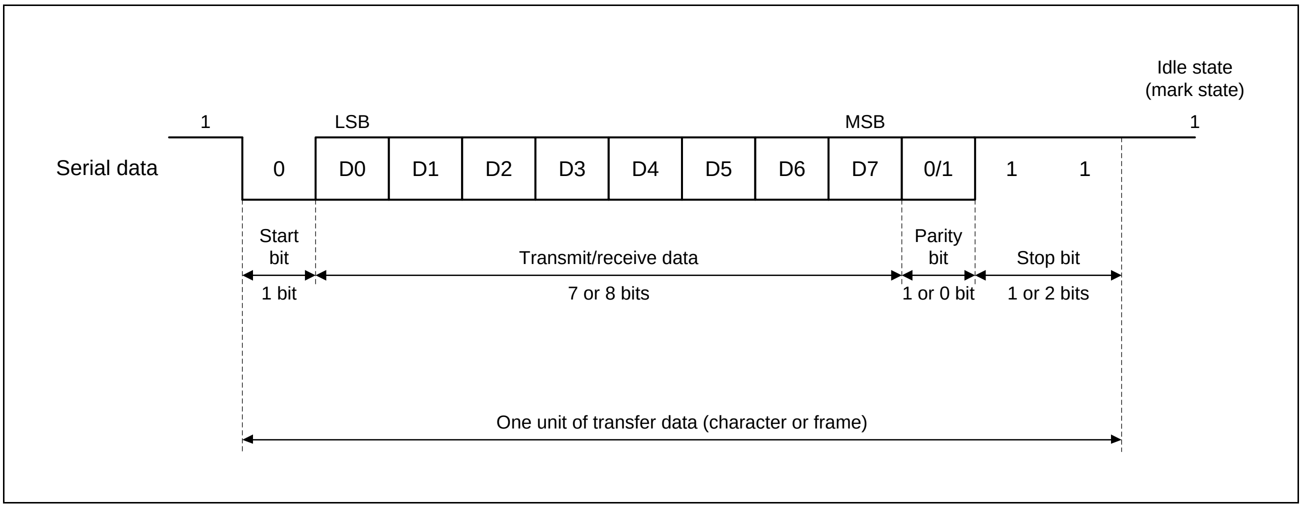


Figure 23.2 Data Format in Asynchronous Serial Communications (Example with 8-Bit Data, Parity, 2 Stop Bits)

23.3.1 Serial Data Transfer Format

Table 23.18 lists the serial data transfer formats that can be used in asynchronous mode.

Any of 18 transfer formats can be selected according to the SMR and SCMR setting. For details of multi-processor function, refer to **Section 23.4, Multi-Processor Communications Function**.

Table 23.18 Serial Transfer Formats (Asynchronous Mode)

SCMR Setting	SMR Setting				Serial Transfer Format and Frame Length												
CHR1	CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	0	0	0	S	9-bit data									STOP		
0	0	0	0	1	S	9-bit data									STOP	STOP	
0	0	1	0	0	S	9-bit data									P	STOP	
0	0	1	0	1	S	9-bit data									P	STOP	STOP
1	0	0	0	0	S	8-bit data								STOP			
1	0	0	0	1	S	8-bit data								STOP	STOP		
1	0	1	0	0	S	8-bit data								P	STOP		
1	0	1	0	1	S	8-bit data								P	STOP	STOP	
1	1	0	0	0	S	7-bit data							STOP				
1	1	0	0	1	S	7-bit data							STOP	STOP			
1	1	1	0	0	S	7-bit data							P	STOP			
1	1	1	0	1	S	7-bit data							P	STOP	STOP		
0	0	—	1	0	S	9-bit data									MPB	STOP	
0	0	—	1	1	S	9-bit data									MPB	STOP	STOP
1	0	—	1	0	S	8-bit data								MPB	STOP		
1	0	—	1	1	S	8-bit data								MPB	STOP	STOP	
1	1	—	1	0	S	7-bit data							MPB	STOP			
1	1	—	1	1	S	7-bit data							MPB	STOP	STOP		

Note: S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multi-processor bit

23.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times*[1] the bit rate.

In reception, the SCI samples the falling edge of the start bit using the base clock, and performs internal synchronization. Since receive data is sampled at the rising edge of the 8th pulse*[1] of the base clock, data is latched at the middle of each bit, as shown in **Figure 23.3**. Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left|(0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\right| \times 100\ [\%] \cdots \text{Formula (1)}$$

M: Reception margin
N: Ratio of bit rate to clock (N = 16 when ABCS in SEMR = 0, N = 8 when ABCS in SEMR = 1)
D: Duty cycle of clock (D = 0.5 to 1.0)
L: Frame length (L = 9 to 13)
F: Absolute value of clock frequency deviation

Assuming values of F = 0 and D = 0.5 in formula (1), the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 16)\} \times 100\ (\%) = 46.875\%$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

Note 1. This is an example when the ABCS bit in SEMR is 0. When the ABCS bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock.

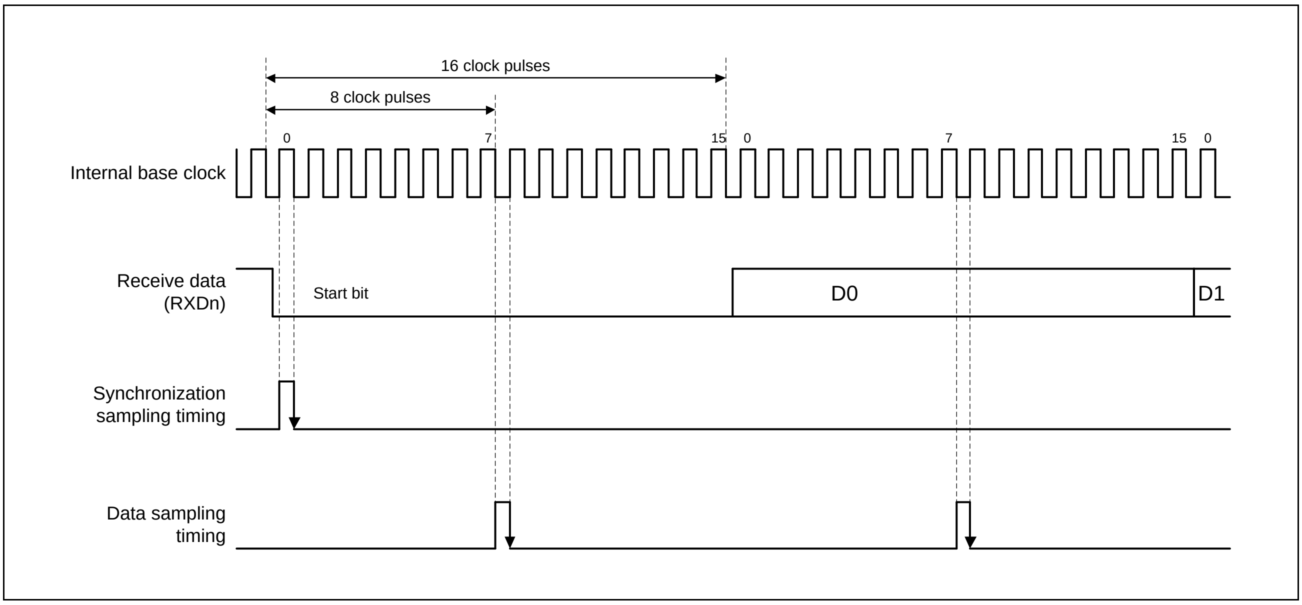


Figure 23.3 Receive Data Sampling Timing in Asynchronous Mode

23.3.3 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input to the SCKn pin can be selected as the SCI's transfer clock, according to the setting of the CM bit in SMR and the CKE[1:0] bits in SCR. When an external clock is input to the SCKn pin, the clock frequency should be 16 times the bit rate (when ABCS in SEMR = 0) and 8 times the bit rate (when ABCS in SEMR = 1).

When the SCI is operated on an internal clock, the clock can be output from the SCKn pin. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in **Figure 23.4**.

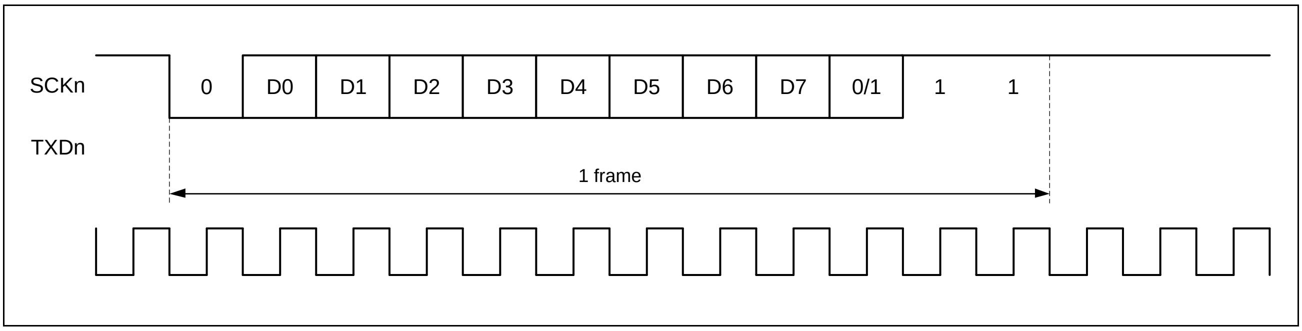


Figure 23.4 Phase Relationship between Output Clock and Transmit Data
(Asynchronous Mode: SMR.CHR = 0, PE = 1, MP = 0, STOP = 1)

23.3.4 Double-Speed Mode

The output clock frequency of the on-chip baud rate generator is doubled by setting the SEMR.BGDM bit to 1, enabling high-speed communication at a doubled bit rate. If the SEMR.ABCS bit is set to 1 under the above condition, the number of base clock cycles changes from 16 to 8, so the bit rate becomes four times faster than the initial state.

As shown by Formula (1) in **Section 23.3.2, Receive Data Sampling Timing and Reception Margin in Asynchronous Mode**, setting the SEMR.ABCS bit to 1 changes the number of cycles to 8, and the sampling interval becomes longer. This causes the reception margin to decrease. Therefore, setting the SEMR.BGDM bit to 1 and the SEMR.ABCS bit to 0 is recommended instead of setting the SEMR.BGDM bit to 0 and the SEMR.ABCS bit to 1 for high-speed operation at a doubled bit rate.

23.3.5 CTS and RTS Functions

The CTS function is the use of input on the CTSn# pin in transmission control. Setting the SECR.CTSE bit to 1 enables the CTS function. When the CTS function is enabled, placing the low level on the CTSn# pin causes transmission to start.

Applying the high level to the CTS# pin while transmission is in progress does not affect transmission of the current frame, which continues.

In the RTS function, by using the function of output on the RTSn# pin, a low level is output when reception becomes possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]

Satisfaction of all conditions listed below

- The value of the RE bit in the SCR is 1
- Reception is not in progress
- There are no received data yet to be read
- The ORER, FER, and PER flags in the SSR are all 0

[Condition for high-level output]

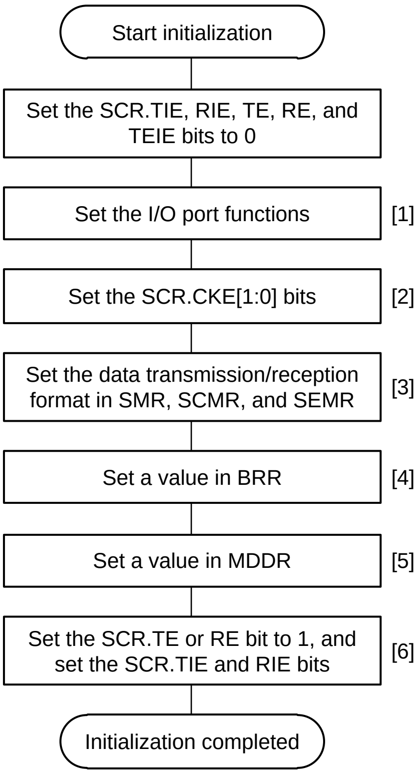
- The conditions for low-level output have not been satisfied.

23.3.6 SCI Initialization (Asynchronous Mode)

Before transmitting and receiving data, start by writing the initial value H’00 to SCR and then continue through the procedure for SCI given in **Figure 23.5**. Whenever the operating mode or transfer format is changed, SCR must be initialized before the change is made.

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization. Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR, RDRH, and RDRL.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.



[1] Make I/O port settings to enable input and output functions as required for TXDn, RXDn, and SCKn pins.

[2] Set the clock selection in SCR.
When the clock output is selected in asynchronous mode, the clock is output immediately after SCR settings are made.

[3] Set data transmission/reception format in SMR, SCMR, and SEMR.

[4] Write a value corresponding to the bit rate to BRR. This step is not necessary if an external clock is used.

[5] Write the value obtained by correcting a bit rate error in MDDR. This step is not necessary if the BRME bit in SEMR is set to 0 or an external clock is used.

[6] Set the SCR.TE or RE bit to 1. Also set the SCR.TIE and RIE bits.
Setting the TE and RE bits allows TXDn and RXDn to be used.

Figure 23.5 Sample SCI Initialization Flowchart (Asynchronous Mode)

23.3.7 Serial Data Transmission (Asynchronous Mode)

Figure 23.6 to **Figure 23.8** show an example of the operation for serial transmission in asynchronous mode. In serial transmission, the SCI operates as described below.

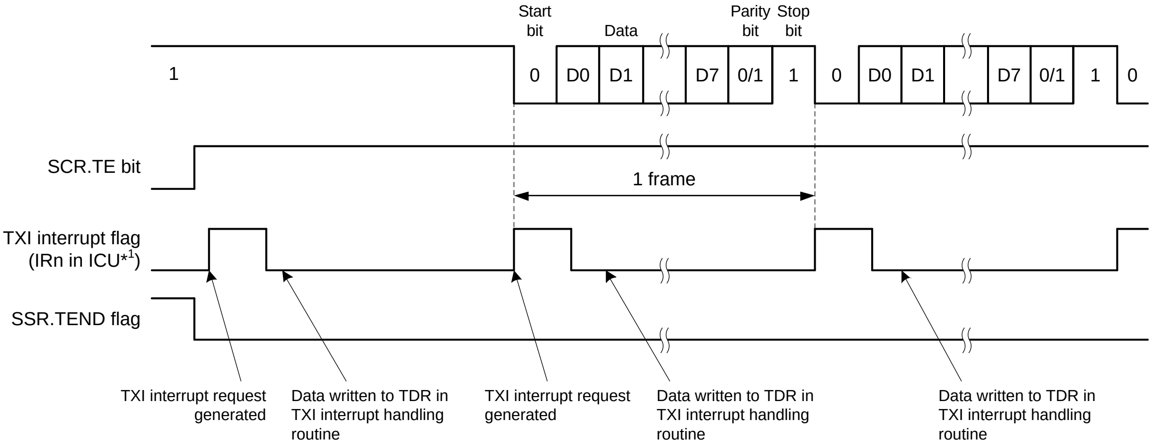
1. The SCI transfers data from TDR*[1] to TSR when data is written to TDR*[1] in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. Transmission starts after the CTSE bit in SECR is set to 0 (CTS function is disabled) and a low level on the CTSn# pin causes data transfer from TDR*[1] to TSR. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Continuous transmission is obtainable by writing the next transmit data to TDR*[1] in the TXI interrupt handling routine before transmission of the current transmit data is completed. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR*[1]*[2] from the handling routine for TXI requests.
3. Data is sent from the TXDn pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks for updating of (writing to) TDR*[3] at the time of stop bit output.
5. When TDR*[3] is updated, setting of the CTSE bit in SECR to 0 (CTS function is disabled) or a low level input on the CTSn# pin cause the next transfer of the next transmit data from TDR*[1] to TSR and sending of the stop bit, after which serial transmission of the next frame starts.
6. If TDR*[3] is not updated, the TEND flag in SSR is set to 1, the stop bit is sent, and then the mark state is entered in which 1 is output. If the TEIE bit in SCR is 1 at this time, the TEND flag in SSR is set to 1 and a TEI interrupt request is generated.

Note 1. Write data not to TDR but to the TDRH and TDRL registers when 9-bit data length is selected.

Note 2. Write data in the order from TDRH to TDRL when 9-bit data length is selected.

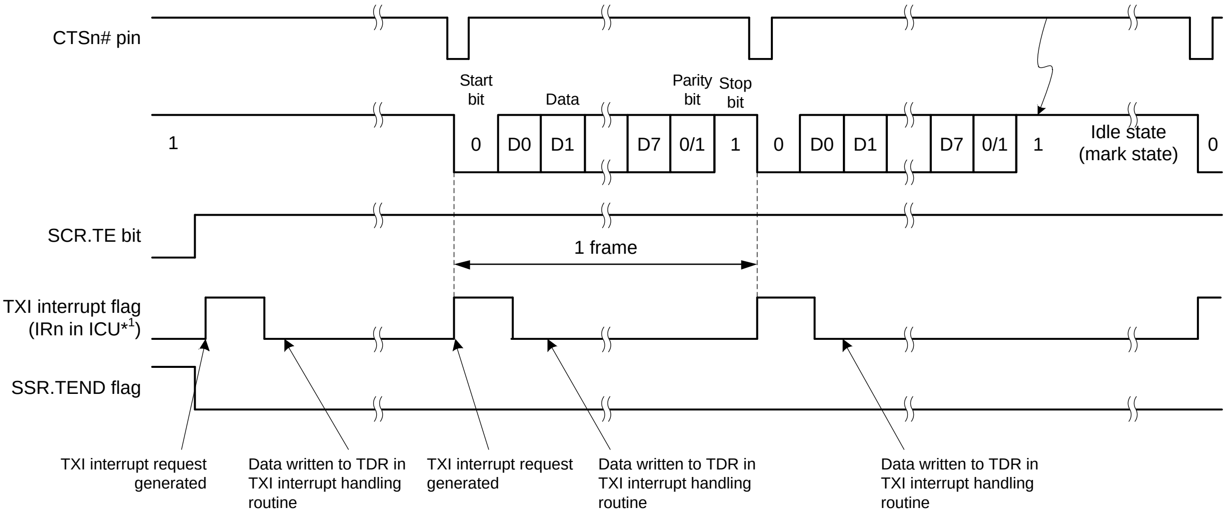
Note 3. The SCI checks for updating of the TDRL register only and does not check for updating of the TDRH register when 9-bit data length is selected.

Figure 23.9 shows a sample flowchart for serial transmission in asynchronous mode.



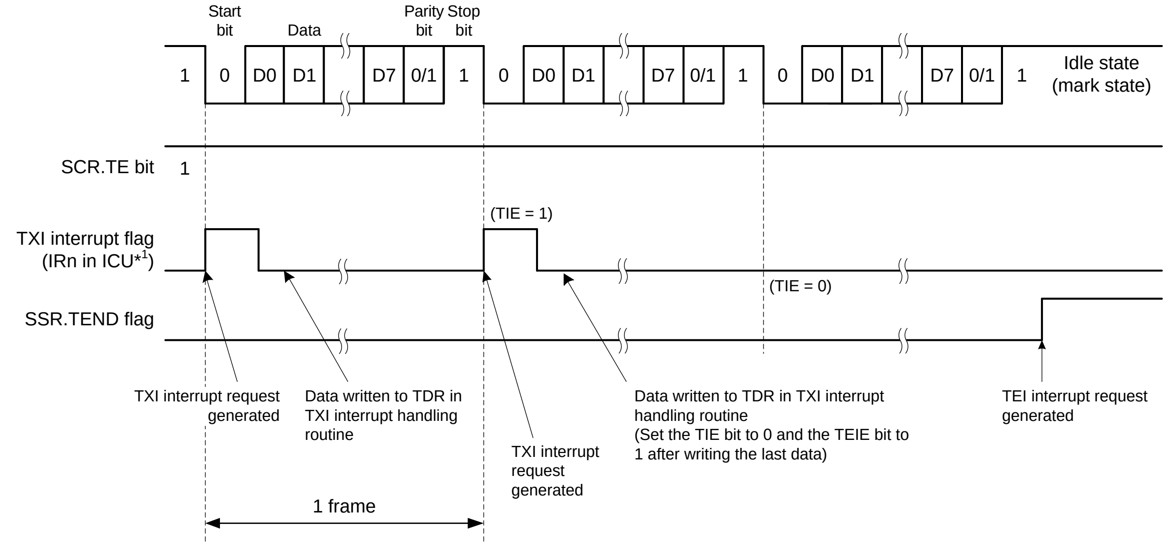
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.6 Example of Operation for Serial Transmission in Asynchronous Mode (1)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, at the Beginning of Transmission)



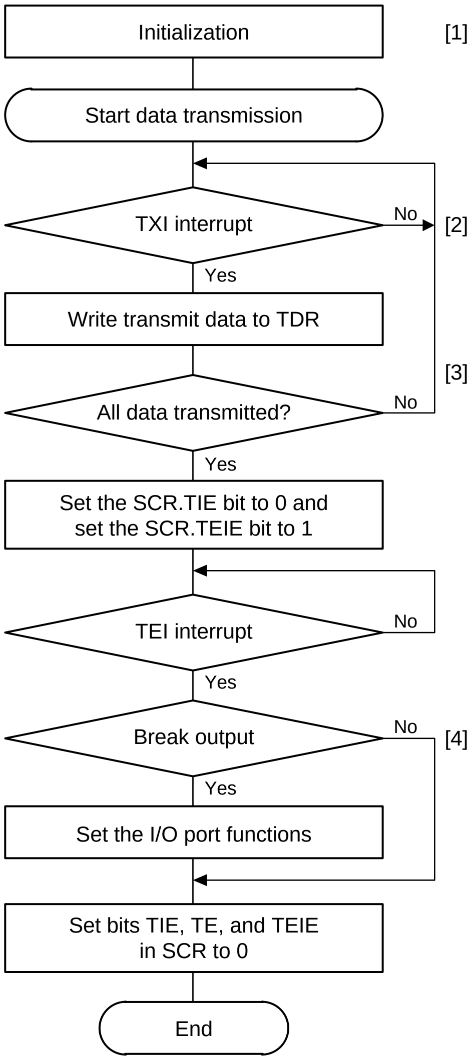
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.7 Example of Operation for Serial Transmission in Asynchronous Mode (2)
(with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Used, at the Beginning of Transmission)



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.8 Example of Operation for Serial Transmission in Asynchronous Mode (3) (with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, from the Middle of Transmission until Transmission Completion)



[1] SCI initialization:
Set data transmission.
After the TE bit in SCR is set to 1, 1 is output for a frame, and transmission is enabled.

[2] Transmit data write to TDR by a TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Write transmit data to TDR once in the TXI interrupt handling routine.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR once using a TXI interrupt request.
Transmit data can also be written to TDR by activating the DMAC.
When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set the I/O port function corresponding to the TXDn pin (corresponding to output of the low level), and after switching the TXDn pin to the general I/O port function, set the TE bit in the SCR to 0.

Note: The TDR register becomes the TDRH and TDRL registers when 9-bit data length is selected. Write data in the order from TDRH to TDRL.

Figure 23.9 Example of Serial Transmission Flowchart in Asynchronous Mode

23.3.8 Serial Data Reception (Asynchronous Mode)

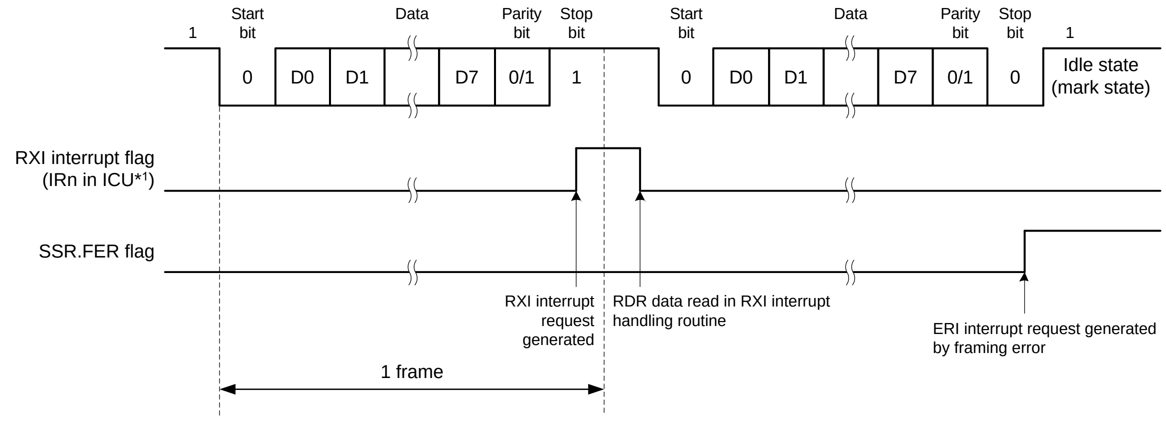
Figure 23.10 and **Figure 23.11** show an example of the operation for serial data reception in asynchronous mode. In serial data reception, the SCI operates as described below.

1. When the value of the RE bit in SCR becomes 1, the output signal on the RTSn# pin goes to the low level.
2. When the SCI monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in RSR, and checks the parity bit and stop bit.
3. If an overrun error occurs, the ORER flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR*[1].
4. If a parity error is detected, the PER bit in SSR is set to 1 and receive data is transferred to RDR*[1]. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
5. If a framing error (when the stop bit is 0) is detected, the FER bit in SSR is set to 1 and receive data is transferred to RDR*[1]. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
6. When reception finishes successfully, receive data is transferred to RDR*[1]. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR*[1] in this RXI interrupt handling routine before reception of the next receive data is completed. Reading the received data that have been transferred to RDR*[1] causes the RTSn# pin to output the low level.

Note 1. Read data not in RDR but in the RDRH and RDRL registers when 9-bit data length is selected.

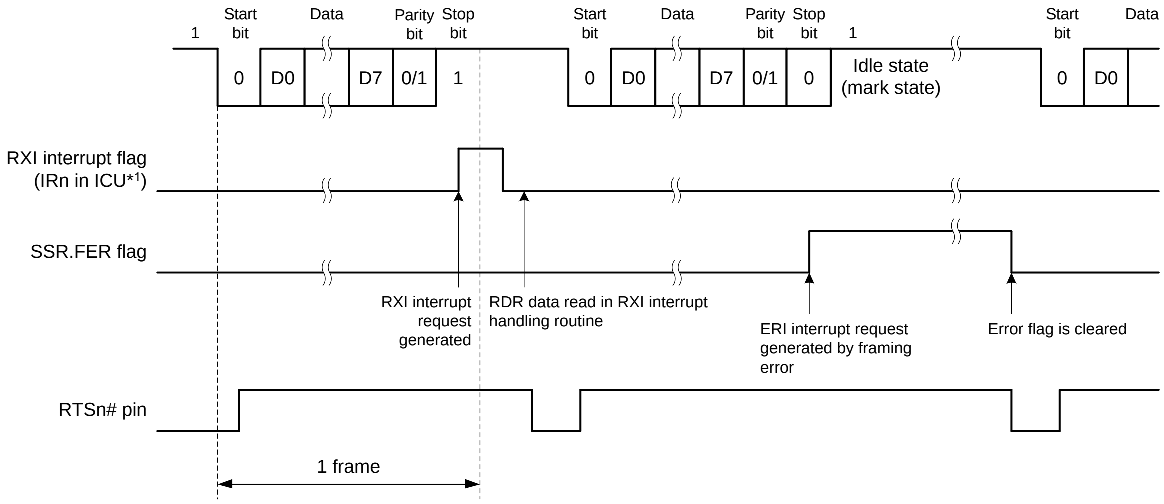
NOTE

The SCI checks for reading of the RDRL register only and does not check for reading of the RDRH register when 9-bit data length is selected.



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.10 Example of SCI Operation for Serial Reception in Asynchronous Mode (1) (When RTS Function is Not Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.11 Example of SCI Operation for Serial Reception in Asynchronous Mode (2) (When RTS Function is Used) (Example with 8-Bit Data, Parity, 1 Stop Bit)

Table 23.19 lists the states of the flags in the SSR status register and receive data handling when a receive error is detected.

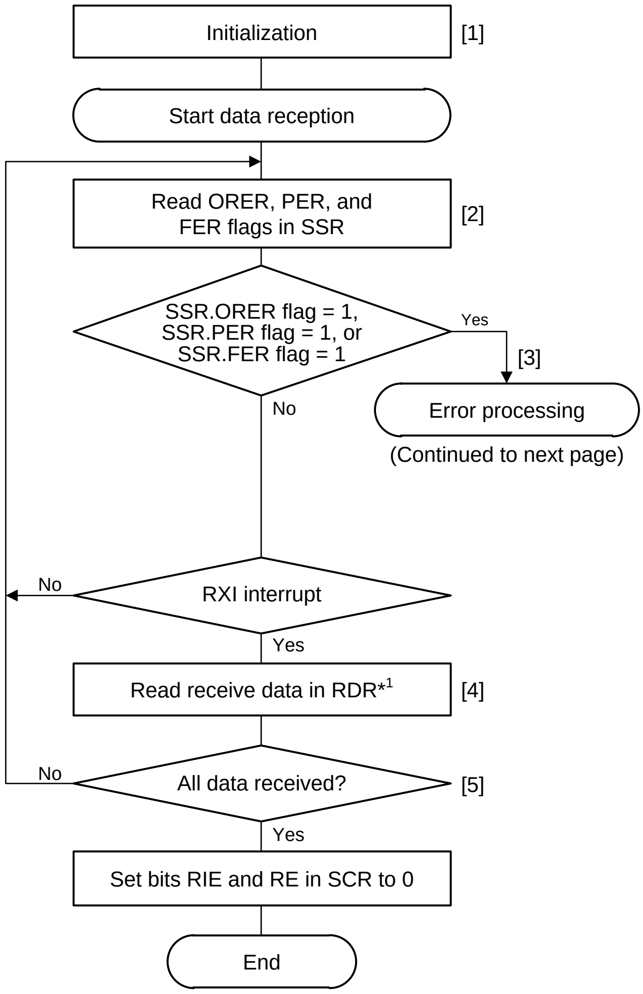
If a receive error is detected, an ERI interrupt request is generated but an RXI interrupt request is not generated. Data reception cannot be resumed while the receive error flag is 1. Accordingly, set the ORER, FER, and PER bits to 0 before resuming reception. Moreover, be sure to read the RDR (or the RDRL) during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read the RDR (or the RDRL) register because received data which has not yet been read may be left in RDR (or the RDRL).

Figure 23.12 and **Figure 23.13** show samples of flowcharts for serial data reception.

Table 23.19 Flags in the SSR Status Register and Receive Data Handling

Flags in the SSR Status Register			Receive Data	Receive Error Type
ORER	FER	PER		
1	0	0	Lost	Overrun error
0	1	0	Transferred to RDR*1	Framing error
0	0	1	Transferred to RDR*1	Parity error
1	1	0	Lost	Overrun error + framing error
1	0	1	Lost	Overrun error + parity error
0	1	1	Transferred to RDR*1	Framing error + parity error
1	1	1	Lost	Overrun error + framing error + parity error

Note 1. Read data not in RDR but in the RDRH and RDRL registers when 9-bit data length is selected.



[1] SCI initialization:
Set data reception.

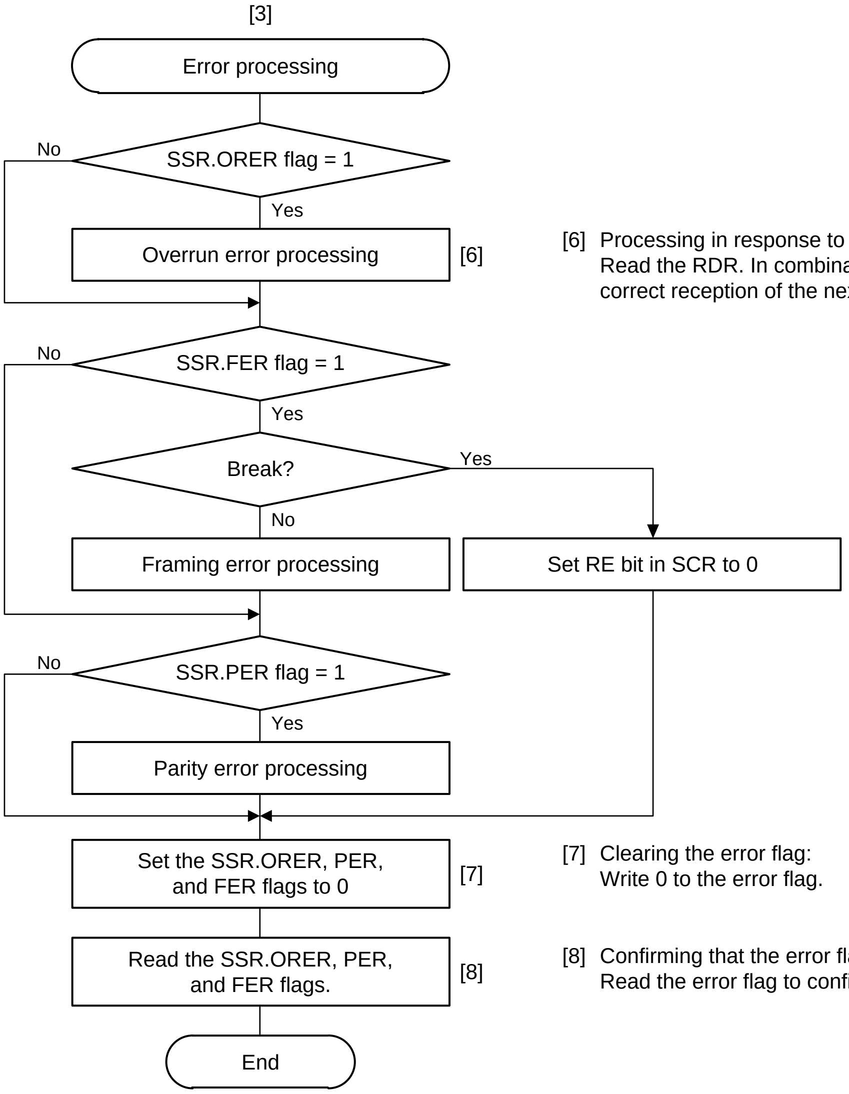
[2][3] Receive error processing and break detection:
If a receive error occurs, an ERI interrupt is generated. An error is identified by reading the ORER, PER, and FER flags in SSR. After performing the appropriate error processing, be sure to set the ORER, PER, and FER flags to 0. Reception cannot be resumed if any of these flags is set to 1. In the case of a framing error, a break can be detected by reading the value of the input port corresponding to the RXDn pin.

[4] Read the receive data in RDR once in the RXI interrupt handling routine.

[5] Serial reception continuation procedure:
To continue serial reception, before the stop bit of the current frame is received, read data from RDR in the RXI interrupt processing routine.
The RDR data can also be read by activating the DMAC.

Note 1. The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 23.12 Example Flowchart of Serial Reception in Asynchronous Mode (1)



[6] Processing in response to an overrun error:
Read the RDR. In combination with step [7], this will make correct reception of the next frame possible

[7] Clearing the error flag:
Write 0 to the error flag.

[8] Confirming that the error flag is cleared:
Read the error flag to confirm that its value is 0.

Note 1. The RDR register becomes the RDRL register when 9-bit data length is selected. Reading the RDRH register is not necessary.

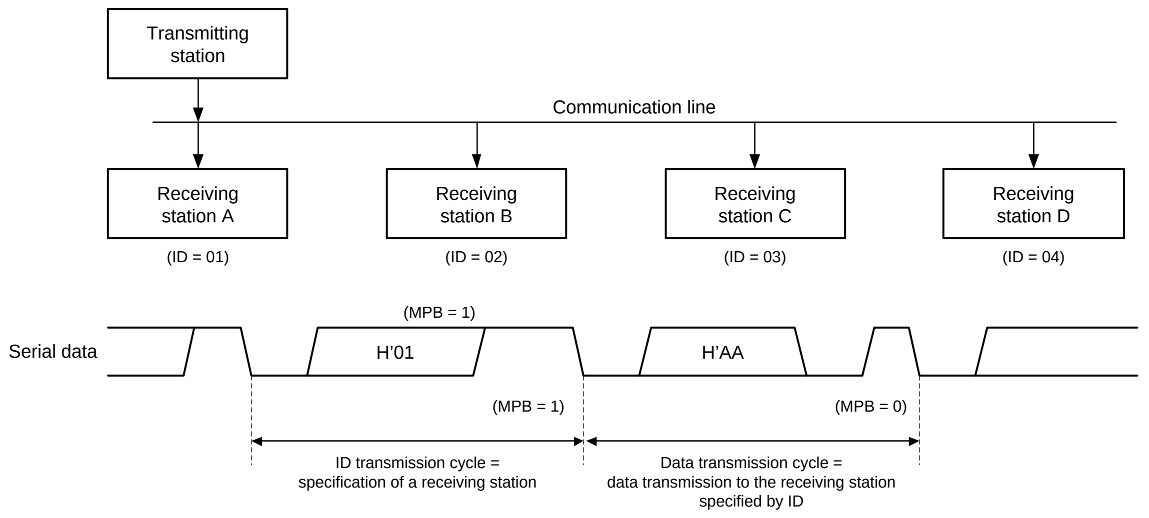
Figure 23.13 Example Flowchart of Serial Reception in Asynchronous Mode (2)

23.4 Multi-Processor Communications Function

Using the multi-processor communication functions enables to transmit and receive data by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish between the ID transmission cycle and the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle and when the multi- processor bit is set to 0, it indicates the data transmission cycle. **Figure 23.14** shows an example of communication between processors by using a multi-processor format. First, a transmitting station transmits communication data in which the multi-processor bit set to 1 is added to the ID code of the receiving station. Next, the transmitting station transmits the communication data in which the multi-processor bit set to 0 is added to the transmit data. Upon receiving the communication data in which the multi-processor bit is set to 1, the receiving station compares the received ID with the ID of the receiving station itself and if the two match, receives the communication data that is subsequently transmitted. If the received ID does not match with the ID of the receiving station, the receiving station skips the communication data until again receiving the communication data in which the multi-processor bit is set to 1.

For supporting this function, the SCI provides the MPIE bit in SCR. When the MPIE bit is set to 1, transfer of receive data from the RSR to the RDR (the RDRH and RDRL registers when 9-bit data length is selected), detection of a receive error, and setting the respective status flags ORER and FER in SSR are disabled until reception of data in which the multi-processor bit is set to 1. Upon receiving a reception character in which the multi-processor bit is set to 1, the MPBT bit in SSR is set to 1 and the MPIE bit in SCR is automatically cleared, thus returning to a normal reception operation. During this time, an RXI interrupt is generated if the RIE bit in SCR is set.

When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the normal asynchronous mode. A clock which is used for the multi-processor communication is also the same as the clock used in the normal asynchronous mode.

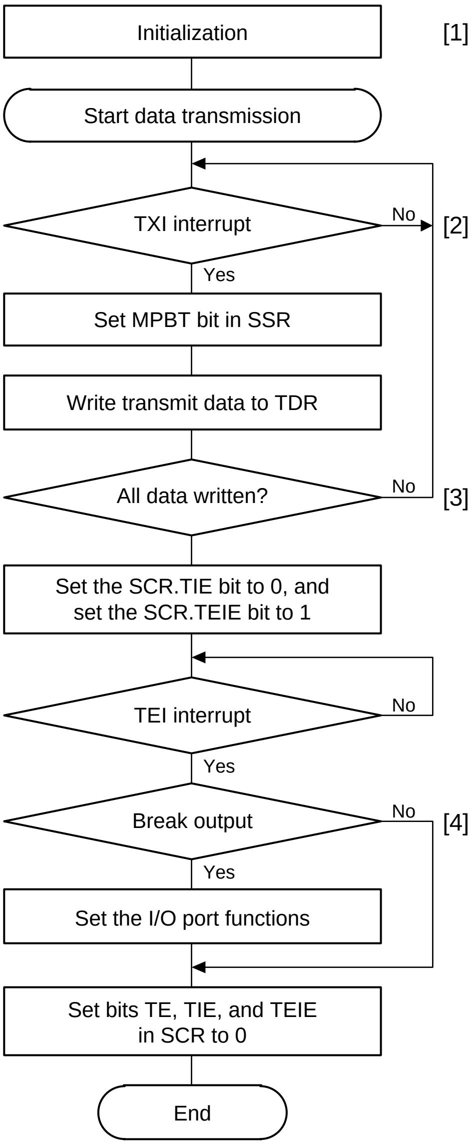


Note: MPB: Multi-processor bit

Figure 23.14 An Example of Communication using the Multi-Processor Format
(Example of Transmission of Data H'AA to Receiving Station A)

23.4.1 Multi-Processor Serial Data Transmission

Figure 23.15 is a sample flowchart of multi-processor data transmission. In the ID transmission cycle, the ID should be transmitted with the MPBT bit in SSR set to 1. In the data transmission cycle, the data should be transmitted with the MPBT bit set to 0. The other operations are the same as the operations in asynchronous mode.



[1] SCI initialization:
Set data transmission.
After the TE bit in SCR is set to 1, 1 is output for a frame, and transmission is enabled.

[2] TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Set the MPBT bit in SSR to 0 or 1, and write transmit data to TDR once in the TXI interrupt handling routine.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR once using a TXI interrupt request.
Transmit data can also be written to TDR by activating the DMAC.
When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set the I/O port function corresponding to the TXDn pin (corresponding to output of the low level), and after switching the TXDn pin to the general I/O port function, set the TE bit in the SCR to 0.

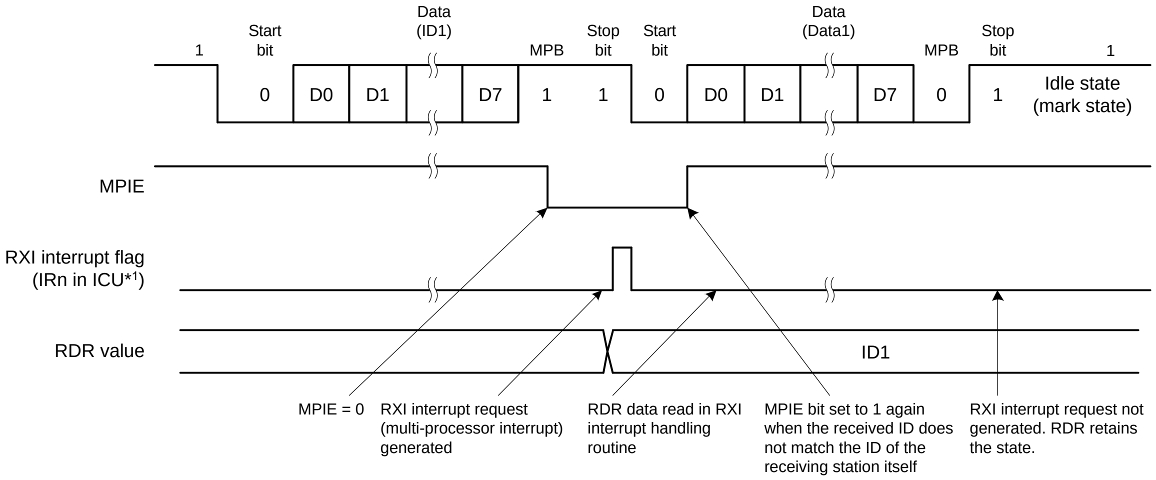
Note: The TDR register becomes the TDRH and TDRL registers when 9-bit data length is selected. Write data in the order from TDRH to TDRL.

Figure 23.15 Example of Multi-Processor Serial Transmission Flowchart

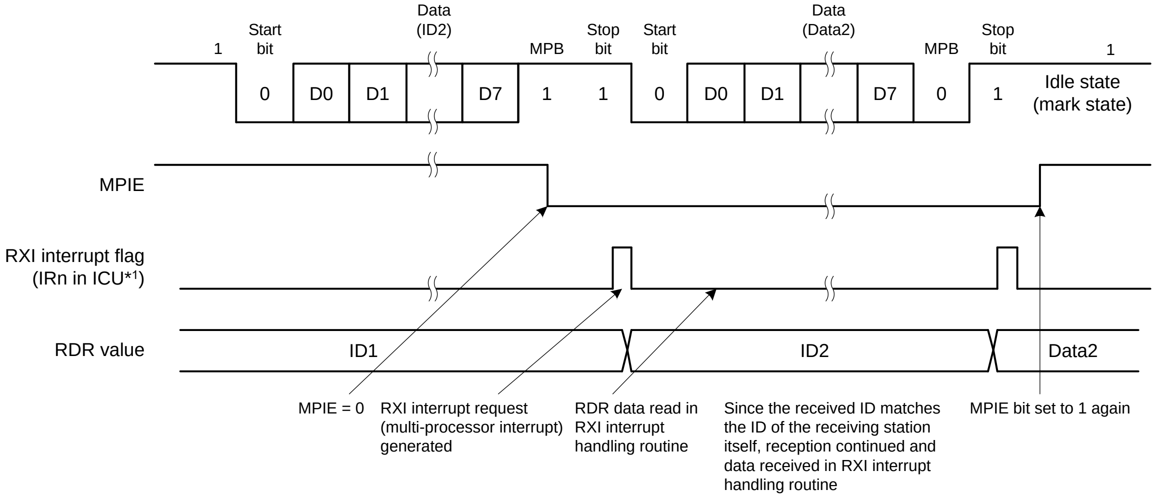
23.4.2 Multi-Processor Serial Data Reception

Figure 23.17 and **Figure 23.18** are sample flowcharts of multi-processor data reception. When the MPIE bit in SCR is set to 1, reading the communication data is skipped until reception of the communication data in which the multi-processor bit is set to 1. When the communication data in which the multi-processor bit is set to 1 is received, the received data is transferred to RDR (the RDRH and RDRL registers when 9-bit data length is selected). During this time, the RXI interrupt request is generated. The other operations are the same as the operations in asynchronous mode.

Figure 23.16 is the example of operation for reception.



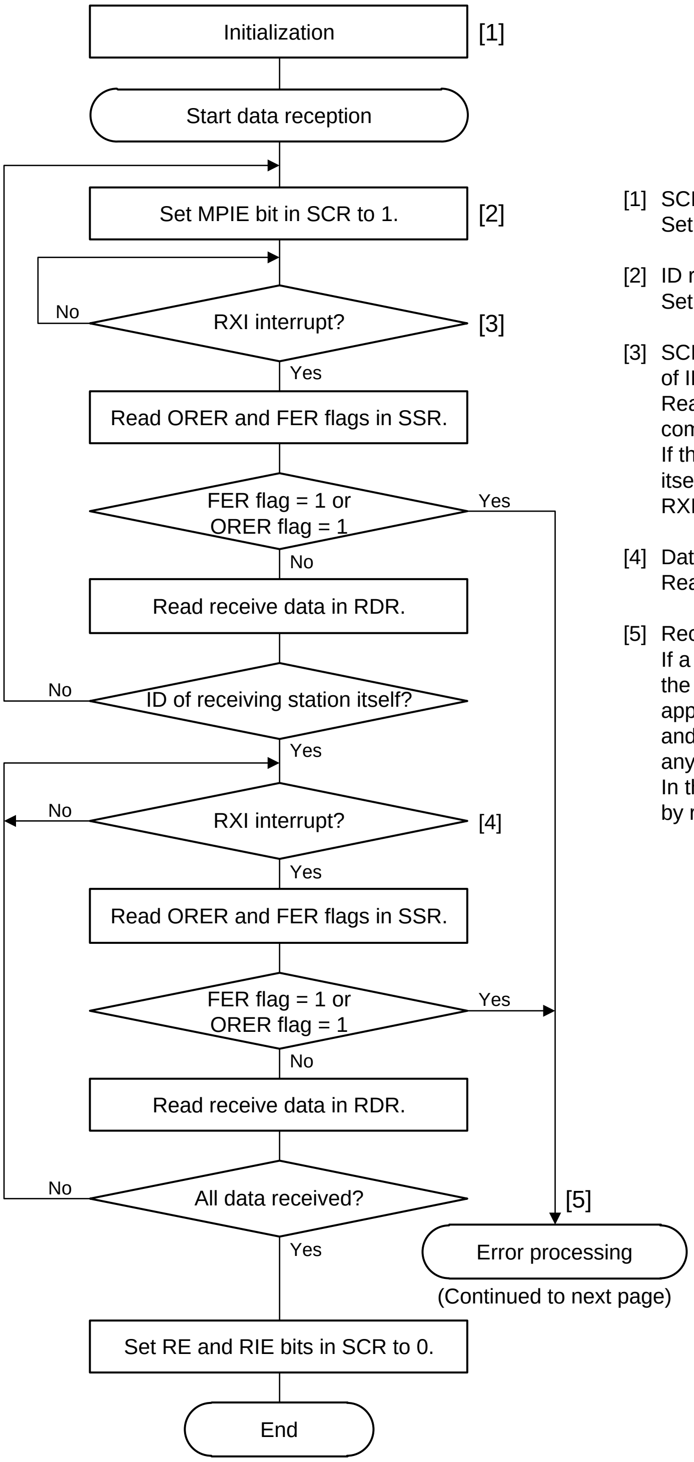
(a) When the received ID does not match the ID of the receiving station itself



(b) When the received ID matches the ID of the receiving station itself

Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.16 Example of SCI Reception (8-Bit Data/Multi-Processor Bit/1 Stop Bit)



[1] SCI initialization:
Set data reception.

[2] ID reception cycle:
Set the MPIE bit in SCR to 1 and wait for ID reception.

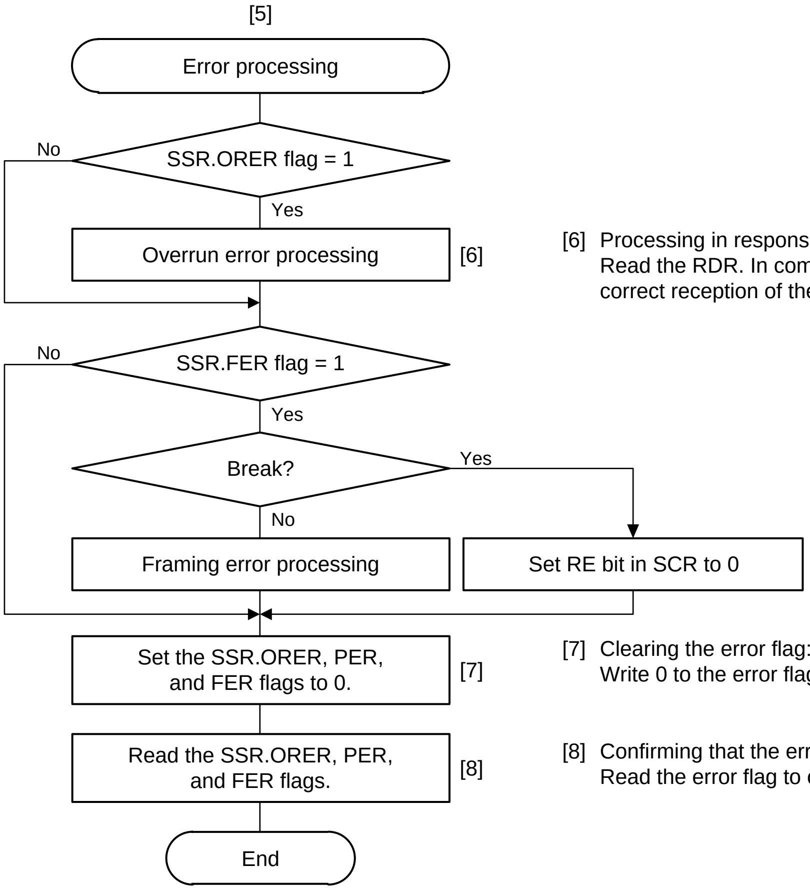
[3] SCI status confirmation and reception and comparison of ID:
Read data in RDR at the first RXI interrupt, and compare it with the ID of the receiving station itself.
If the ID does not match the ID of the receiving station itself, set the MPIE bit to 1 again, and wait for another RXI interrupt RXI interrupt request.

[4] Data reception at an RXI interrupt:
Read data in RDR once in the RXI interrupt routine.

[5] Receive error processing and break detection:
If a receive error occurs, an error is identified by reading the ORER and FER flags in SSR. After performing the appropriate error processing, be sure to set the ORER and FER flags to 0. Reception cannot be resumed if any of these flags is set to 1.
In the case of a framing error, a break can be detected by reading the value of the RXDn pin.

Note: The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 23.17 Example of Multi-Processor Serial Reception Flowchart (1)



[6] Processing in response to an overrun error:
Read the RDR. In combination with step [7], this will make correct reception of the next frame possible.

[7] Clearing the error flag:
Write 0 to the error flag.

[8] Confirming that the error flag is cleared:
Read the error flag to confirm that its value is 0.

Note: The RDR register becomes the RDRH and RDRL registers when 9-bit data length is selected. Read data in the order from RDRH to RDRL.

Figure 23.18 Example of Multi-Processor Serial Reception Flowchart (2)

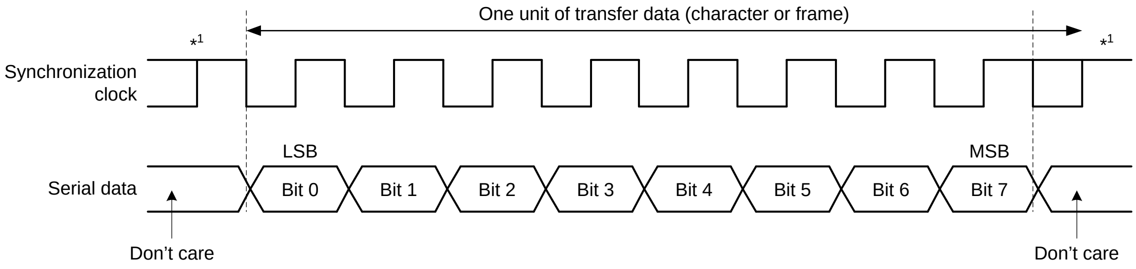
23.5 Operation in Clock Synchronous Mode

Figure 23.19 shows the data format for clock synchronous serial data communications.

In clock synchronous mode, data is transmitted or received in synchronization with clock pulses. One character in transfer data consists of 8-bit data. In clock synchronous mode, no parity bit can be added.

In data transmission, the SCI outputs data from one falling edge of the synchronization clock to the next. In data reception, the SCI receives data in synchronization with the rising edge of the synchronization clock. After 8-bit data is output, the communication line holds the last bit output state.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications by use of a common clock. Both the transmitter and the receiver also have a double-buffered structure, so that the next transmit data can be written during transmission or the previous receive data can be read during reception, enabling continuous data transfer.



Note 1. Holds a high level except during continuous transfer.

Figure 23.19 Data Format in Clock Synchronous Serial Communications (LSB First)

23.5.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCKn pin can be selected, according to the setting of the SCR.CKE[1:0] bits.

When the SCI is operated on an internal clock, the synchronization clock is output from the SCKn pin. Eight synchronization clock pulses are output in the transfer of one character, and when no transfer is performed the clock is held high. However, when only data reception is performed while the CTS function is disabled, the synchronization clock output is started at the same time when the SCR.RE bit set to 1. The synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.

When only data reception is performed and the CTS function is enabled, the clock output is not started even when the SCR.RE bit set to 1 if the CTSn# pin input is high when the SCR.RE bit is 0. The synchronization clock output is started when the SCR.RE bit is set to 1 and the CTSn# pin input is low. After that, if the CTSn# pin input is high on completion of the frame reception, the synchronization clock output is stopped at the high level. If the CTSn# pin input continues to be low, the synchronization clock is stopped at the high level when an overrun error occurs or the SCR.RE bit is set to 0.

23.5.2 CTS and RTS Functions

In the CTS function, CTSn# pin input is used to control reception/transmission start when the clock source is the internal clock. Setting the SECR.CTSE bit to 1 enables the CTS function.

When the CTS function is enabled, placing the low level on the CTSn# pin causes reception/transmission to start.

In the RTS function, RTSn# pin output is used to request reception/transmission start when the clock source is an external synchronizing clock. A low level is output when serial communications become possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output] Satisfaction of all conditions listed below

- The value of the RE or TE bit in the SCR is 1
- Neither transmission nor reception is in progress
- There are no received data yet to be read (when the SCR.RE bit is 1)
- Transmit data has been written (when the SCR.TE bit is 1)
- ORER flag in SSR is 0

[Condition for high-level output]

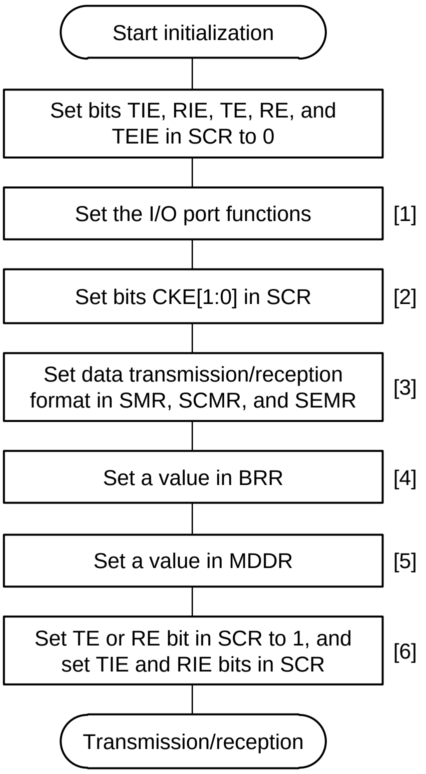
- The conditions for low-level output have not been satisfied.

23.5.3 SCI Initialization (Clock Synchronous Mode)

Before transmitting and receiving data, start by writing the initial value H'00 to the SCR and then continue through the procedure for SCI given in **Figure 23.20**. Whenever the operating mode or transfer format is changed, the SCR must be initialized before the change is made.

Note that setting the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.



[1] Make I/O port settings to enable input and output functions as required for TXDn, RXDn, and SCKn pins.

[2] Set the clock selection in SCR.
When an internal clock is selected, the SCK pin functions as the clock output pin.

[3] Set the data transmission/reception format in SMR, SCMR, and SEMR.

[4] Write a value corresponding to the bit rate to BRR. This step is not necessary if an external clock is used.

[5] Write the value obtained by correcting a bit rate error in MDDR. This step is not necessary if the BRME bit in SEMR is cleared to 0 or an external clock is used.

[6] Set the TE bit or RE bit in SCR to 1. Also set the TIE and RIE bits in SCR.
Setting the TE and RE bits allows TXDn and RXDn to be used.

Note: In simultaneous transmit and receive operations, the TE and RE bits in SCR should both be set to 0 or 1 simultaneously.

Figure 23.20 Example of SCI Initialization Flowchart (Clock Synchronous Mode)

23.5.4 Serial Data Transmission (Clock Synchronous Mode)

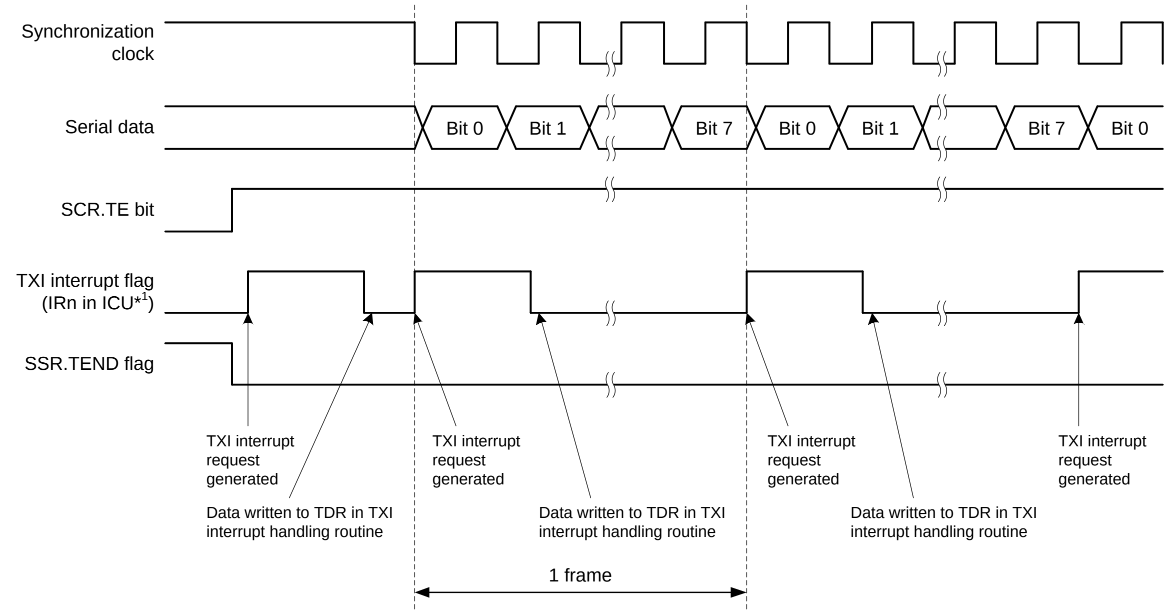
Figure 23.21, **Figure 23.22**, and **Figure 23.23** show an example of the operation for serial transmission in clock synchronous mode.

In serial data transmission, the SCI operates as described below.

1. The SCI transfers data from TDR to TSR when data is written to TDR in the TXI interrupt handling routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these 2 bits are set to 1 simultaneously by a single instruction.
2. After transferring data from TDR to TSR, the SCI starts transmission. When the SCR.TIE bit is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to TDR in this TXI interrupt handling routine before transmission of the current transmit data has finished. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (a TXI interrupt request is disabled) and the SCR.TEIE bit to 1 (a TEI interrupt request is enabled) after the last of the data to be transmitted are written to the TDR from the handling routine for TXI requests.
3. 8-bit data is sent from the TXDn pin in synchronization with the output clock when clock output mode has been specified and in synchronization with the input clock when use of an external clock has been specified. Output of the clock signal is suspended until the input CTS signal is at the low level while the CTSE bit in SECR is 1 (CTS function is enabled).
4. The SCI checks for updating of (writing to) the TDR at the time of the last bit output.
5. When TDR is updated, the next transmit data is transferred from TDR to TSR, and serial transmission of the next frame is started.
6. If TDR is not updated, set the SSR.TEND flag to 1 and the TXDn pin retains the output state of the last bit. If the TEIE bit in SCR is 1 at this time, a TEI interrupt request is generated. The SCKn pin is held high.

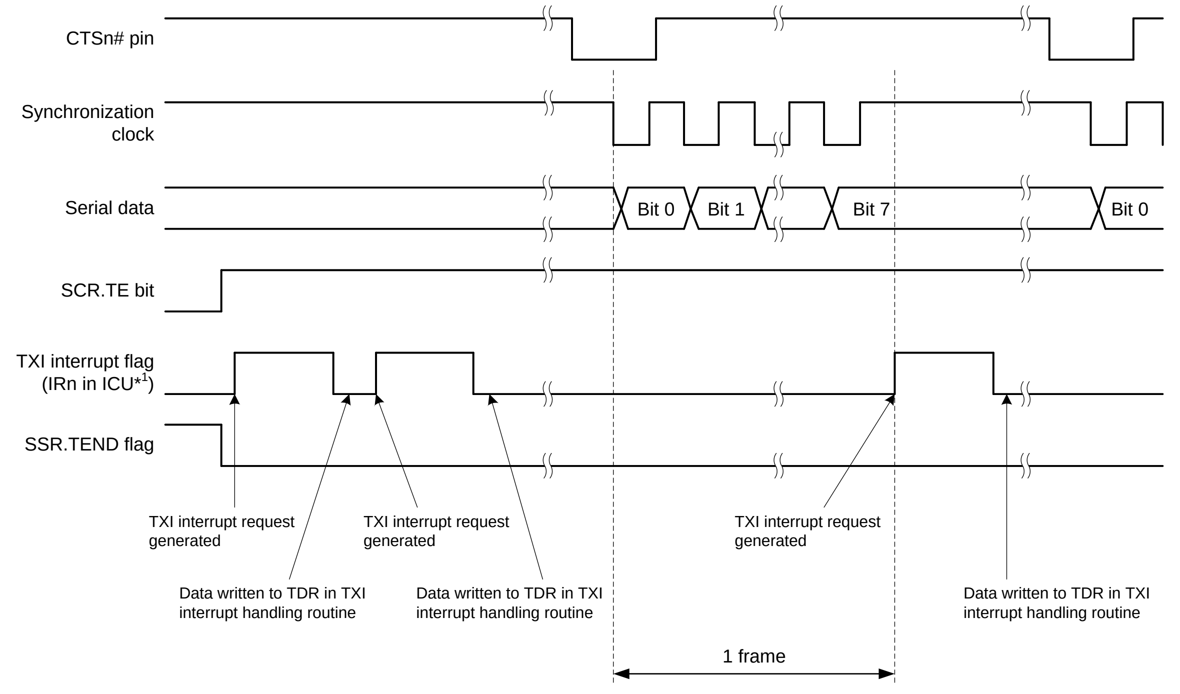
Figure 23.24 shows a sample flowchart of serial data transmission.

Transmission will not start while a receive error flag (ORER, FER, or PER in SSR) is set to 1. Be sure to set the receive error flags to 0 before starting transmission. Note that setting the RE bit in SCR to 0 does not clear the receive error flags.



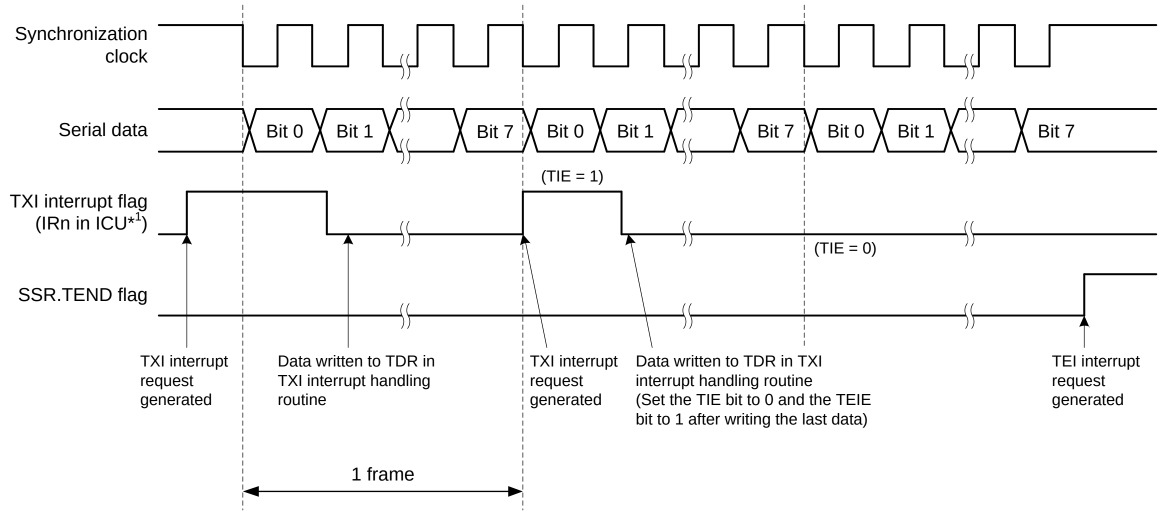
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.21 Example of Serial Data Transmission in Clock Synchronous Mode (1)
(When the CTS Function is Not Used at the Beginning of Transmission)



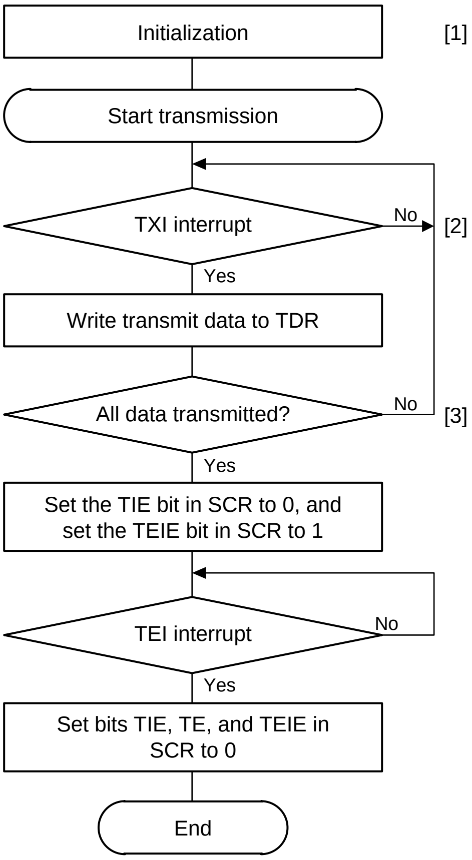
Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.22 Example of Serial Data Transmission in Clock Synchronous Mode (2) (When the CTS Function is Used at the Beginning of Transmission)



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.23 Example of Serial Data Transmission in Clock Synchronous Mode (3)
(From the Middle of Transmission until Transmission Completion)



[1] SCI initialization:
Set data transmission.

[2] Writing transmit data write to TDR by a TXI interrupt request:
When transmit data is transferred from TDR to TSR, a transmit data empty interrupt (TXI) request is generated.
Transmit data is written to TDR once from the handling routine for TXI requests.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to TDR upon accepting a transmit data empty interrupt (TXI) request. Transmit data can also be written to TDR by activating the DMAC by the TXI interrupt request. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 and the SCR.TEIE bit to 1 after the last of the data to be transmitted are written to the TDR.

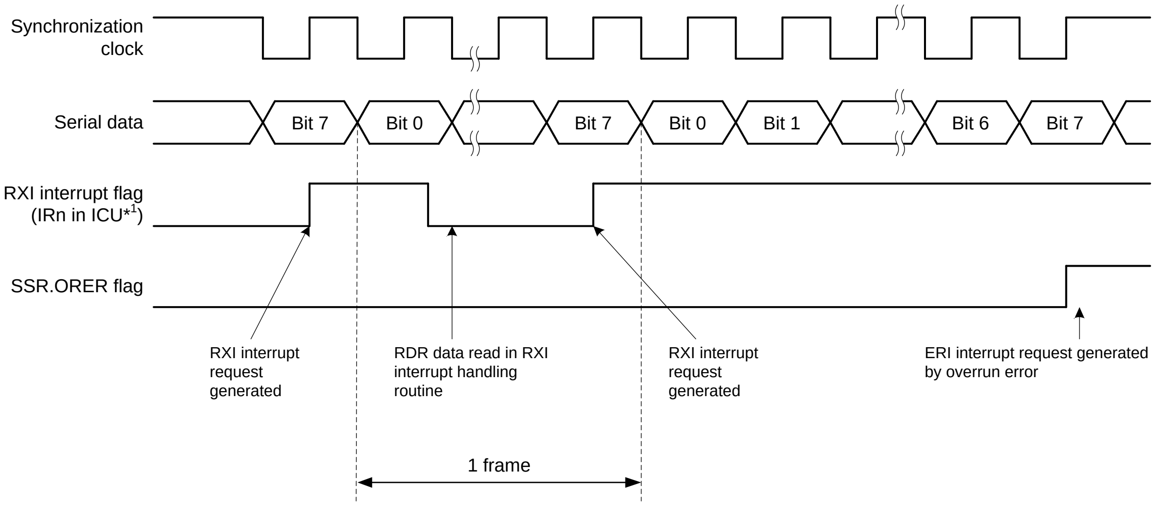
Note: When the external clock is in use (the value of the SCR.CKE[1:0] bits is 10b or 11b), the rising edge on the SCK pin for the last bit sets the SSR.TEND flag to 1. Setting the SCR.TE bit to 0 immediately after this may lead to insufficient received-data hold time on the receiver side.

Figure 23.24 Example Flowchart of Serial Transmission in Clock Synchronous Mode

23.5.5 Serial Data Reception (Clock Synchronous Mode)

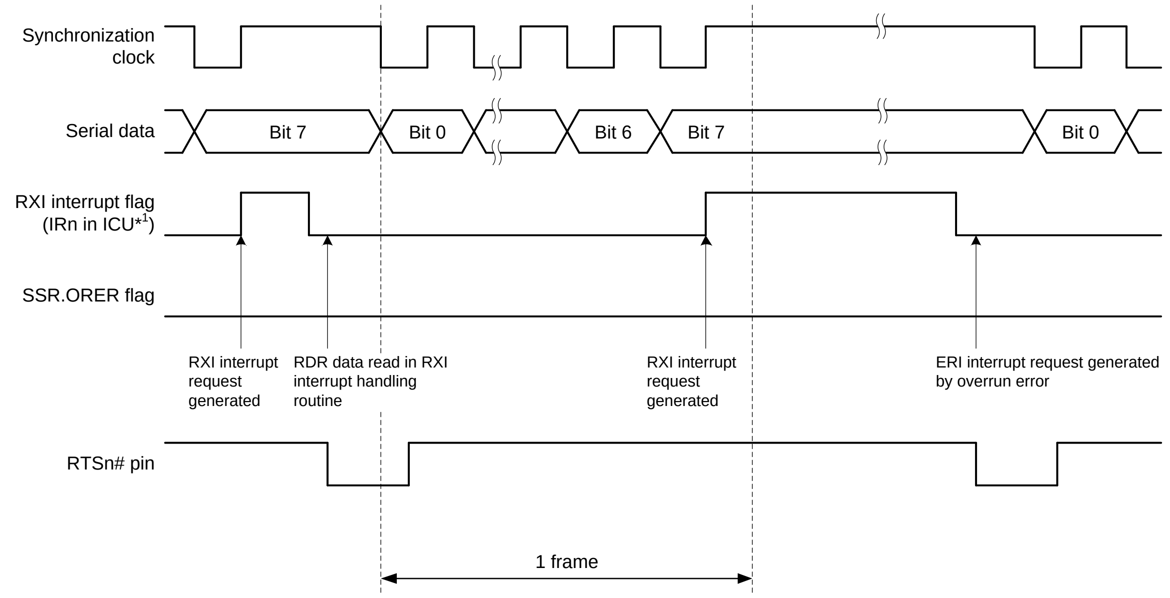
Figure 23.25 and **Figure 23.26** show an example of SCI operation for serial reception in clock synchronous mode. In serial data reception, the SCI operates as described below.

1. The value of the RE bit in SCR becoming 1 places the signal output on the RTSn# pin at the low level (when the RTS function is in use).
2. The SCI performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores the receive data in RSR.
3. If an overrun error occurs, the ORER bit in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR.
4. When reception finishes successfully, receive data is transferred to RDR. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR in this RXI interrupt handling routine before reception of the next receive data is completed. Reading out the received data that have been transferred to RDR causes the RTSn# pin to output the low level (when the RTS function is in use).



Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.25 Example of Operation for Serial Reception in Clock Synchronous Mode (1) (When RTS Function is Not Used)

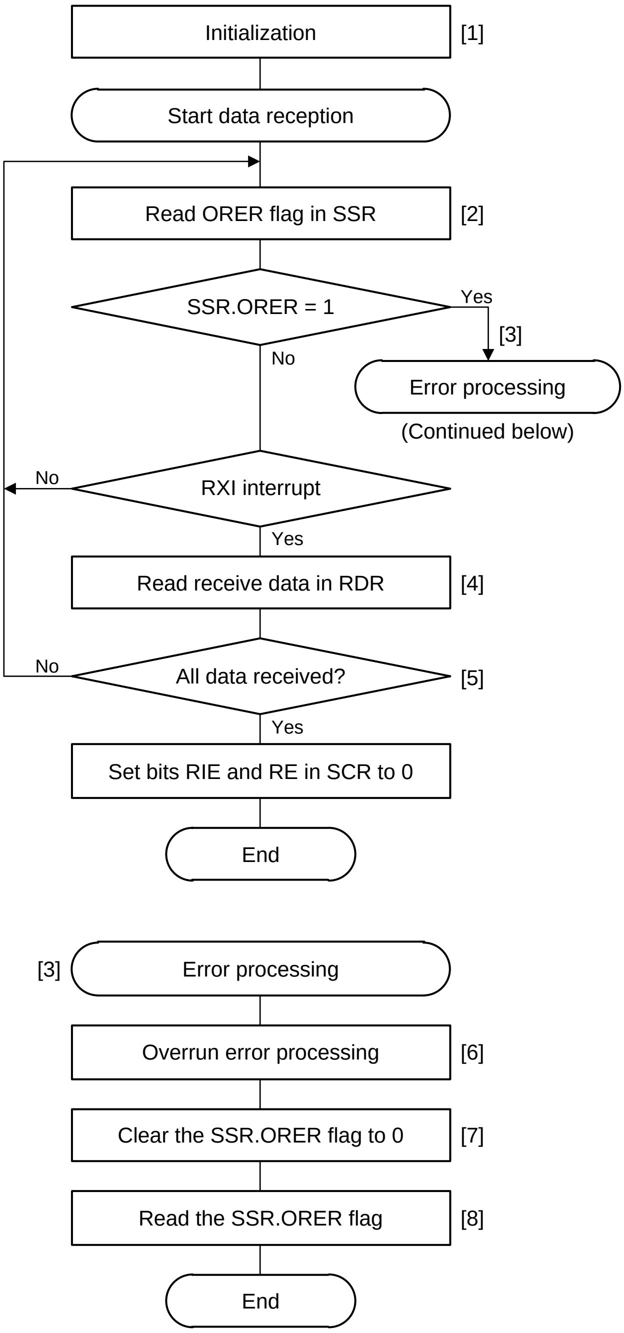


Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.26 Example of Operation for Serial Reception in Clock Synchronous Mode (2) (When RTS Function is Used)

Data transfer cannot be resumed while a receive error flag is 1. Accordingly, clear the ORER, FER, and PER bits in SSR to 0 before resuming reception. Moreover, be sure to read the RDR during overrun error processing. When a reception is forcibly terminated by setting the SCR.RE bit to 0 during operation, read RDR because received data which has not yet been read may be left in RDR.

Figure 23.27 shows a sample flowchart for serial data reception.



[1] SCI initialization:
Make input port-pin settings for pins to be used as RXDn pins.

[2][3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, perform the relevant error processing, and then set the ORER flag to 0. Data reception cannot be resumed while the ORER flag is 1.

[4] Read the receive data in RDR once in the receive data full interrupt (RXI) request handling routine.

[5] Serial reception continuation procedure:
To continue serial reception, before the MSB (bit 7) of the current frame is received, finish reading the receive data in RDR. The RDR data can also be read by activating the DMAC by an RXI interrupt request.

[6] Processing in response to an overrun error: Read the RDR. In combination with step [7], this will make correct reception of the next frame possible.

[7] Clearing the error flag:
Write 0 to the error flag.

[8] Confirming that the error flag is cleared:
Read the error flag to confirm that its value is 0.

Figure 23.27 Example Flowchart of Serial Reception in Clock Synchronous Mode

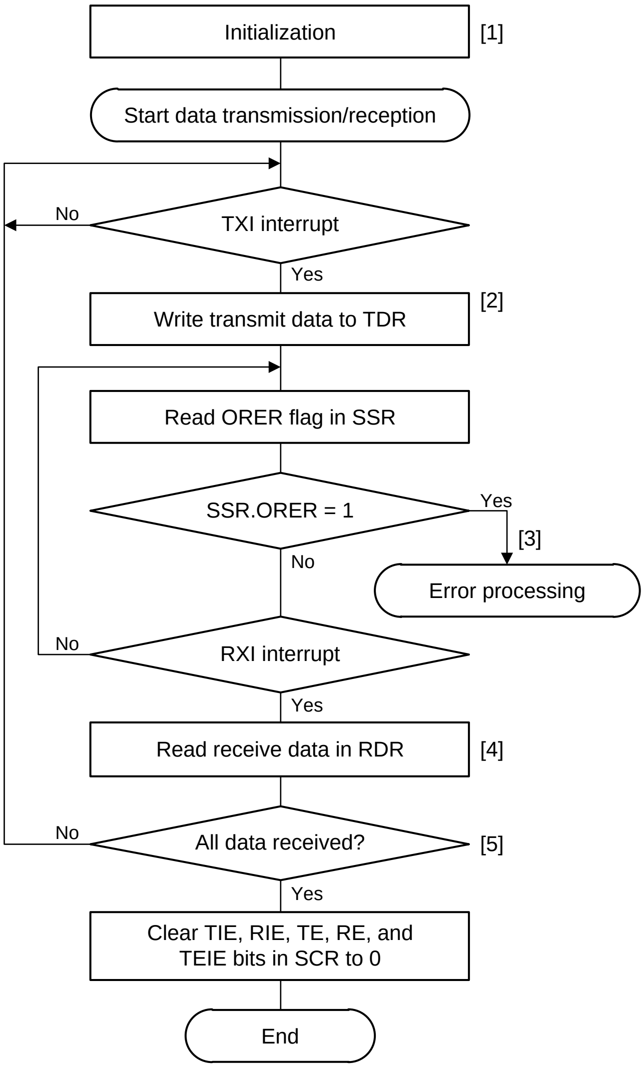
23.5.6 Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)

Figure 23.28 shows a sample flowchart for simultaneous serial transmit and receive operations in clock synchronous mode.

After initializing the SCI, the following procedure should be used for simultaneous serial data transmit and receive operations.

To switch from transmit mode to simultaneous transmit and receive mode, check that the SCI has finished transmission by reading that the TEND flag in SSR is 1, and then initialize the SCR register. Then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.

To switch from receive mode to simultaneous transmit and receive mode, check that the SCI has finished reception, and then set the RIE and RE bits to 0. Then check that the receive error flags (ORER, FER, and PER in SSR) are 0, and then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.



[1] SCI initialization:
The TXDn pin can act as the output pin for transmitted data and the RXDn pin can act as the input pin for received data at the same time.

[2] Transmit data write:
Write transmit data to TDR once in the TXI interrupt handling routine.

[3] Receive error processing:
If a receive error occurs, read the ORER flag in SSR, perform the relevant error processing, and then set the ORER flag to 0. Data reception cannot be resumed while the ORER flag is 1.

[4] Reading receive data:
Read the receive data in RDR once in the RXI interrupt request handling routine.

[5] Serial transmission/reception continuation procedure:
To continue serial transmission and reception, before the MSB (bit 7) of the current frame is received, finish reading the receive data in RDR by the RXI interrupt. Also, before the MSB (bit 7) of the current frame is transmitted, write data to TDR by the TXI interrupt.
Transmit data can also be written to TDR by activating the DMAC by a transmit data empty interrupt (TXI) request. Similarly, the RDR data can also be read by activating the DMAC by a receive data full interrupt (RXI) request.

Note: When switching from transmit or receive operation to simultaneous transmit and receive operations first set the TIE, RIE, TE, RE, and TEIE bits in SCR to 0, and then set TIE, RIE, TE, and RE bits to 1 simultaneously.

Figure 23.28 Example Flowchart of Simultaneous Serial Transmission and Reception in Clock Synchronous Mode

23.6 Operation in Smart Card Interface Mode

The SCI supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards), as an extended function of the SCI.

Smart card interface mode can be selected using the appropriate register.

23.6.1 Sample Connection

Figure 23.29 shows a sample connection between a smart card (IC card) and this MCU.

As in the figure, since this MCU communicates with an IC card using a single transmission line, interconnect the TXDn and RXDn pins and pull up the data transmission line to VCC using a resistor.

Setting the TE and RE bits in SCR to 1 with an IC card disconnected enables closed transmission/reception allowing self-diagnosis.

To supply an IC card with the clock pulses generated by the SCI, input the SCKn pin output to the CLK pin of an IC card. The output port of this MCU can be used to output a reset signal.

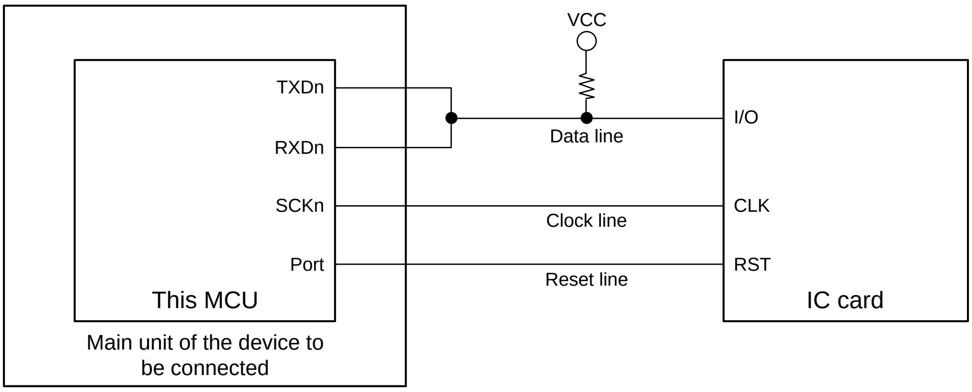
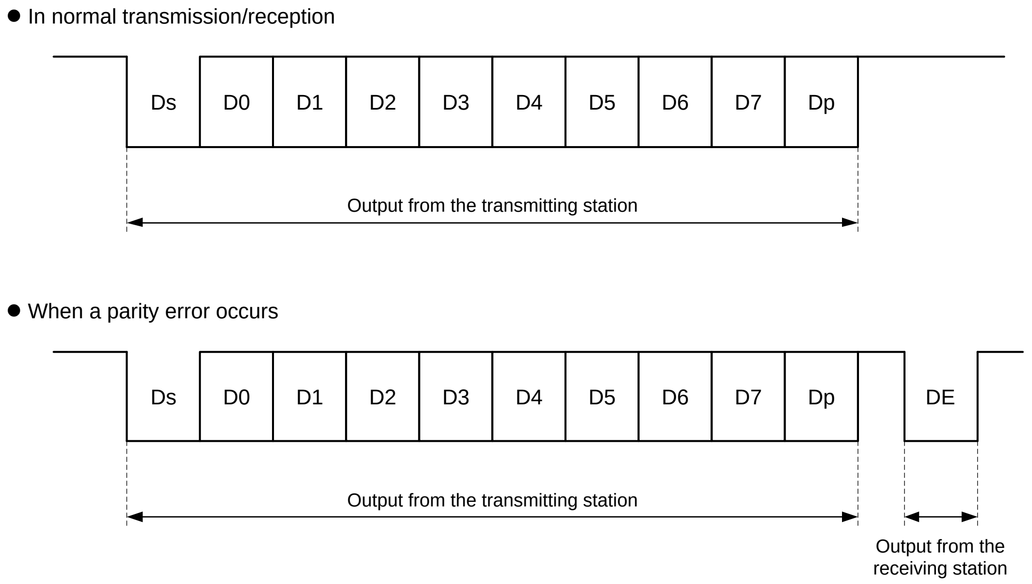


Figure 23.29 Sample Connection with a Smart Card (IC Card)

23.6.2 Data Format (Except in Block Transfer Mode)

Figure 23.30 shows the data transfer formats in smart card interface mode.

- One frame consists of 8-bit data and a parity bit in asynchronous mode.
- During transmission, at least 2 etu (elementary time unit: time required for transferring 1 bit) is secured as a guard time from the end of the parity bit until the start of the next frame.
- If a parity error is detected during reception, a low-level error signal is output for 1 etu after 10.5 etu has passed from the start bit.
- If an error signal is sampled during transmission, the same data is automatically retransmitted after at least 2 etu.



Note: Ds: Start bit
D0 to D7: Data bits
Dp: Parity bit
DE: Error signal

Figure 23.30 Data Formats in Smart Card Interface Mode

For communications with IC cards of the direct convention type and inverse convention type, follow the procedure below.

(1) Direct Convention Type

For the direct convention type, logic levels 1 and 0 correspond to states Z and A, respectively, and data is transferred with LSB first as the start character, as shown in **Figure 23.31**. Therefore, data in the start character in the figure is H’3B.

When using the direct convention type, write 0 to both the SDIR and SINV bits in SCMR. Write 0 to the PM bit in SMR in order to use even parity, which is prescribed by the smart card standard.

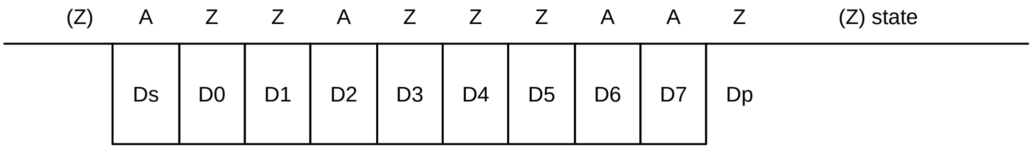


Figure 23.31 Direct Convention (SDIR in SCMR = 0, SINV in SCMR = 0, PM in SMR = 0)

(2) Inverse Convention Type

For the inverse convention type, logic levels 1 and 0 correspond to states A and Z, respectively and data is transferred with MSB first as the start character, as shown in **Figure 23.32**. Therefore, data in the start character in the figure is H’3F. When using the inverse convention type, write 1 to both the SDIR and SINV bits in SCMR. The parity bit is logic level 0 to produce even parity, which is prescribed by the smart card standard, and corresponds to state Z. Since the SINV bit of this MCU only inverts data bits D7 to D0, write 1 to the PM bit in SMR to invert the parity bit for both transmission and reception.

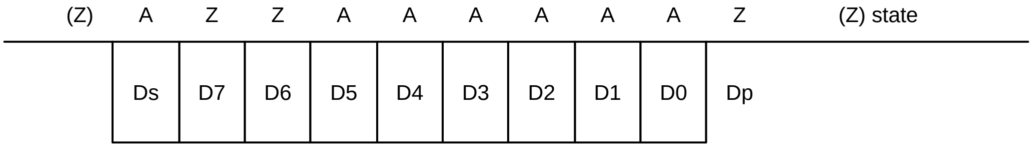


Figure 23.32 Inverse Convention (SDIR in SCMR = 1, SINV in SCMR = 1, PM in SMR = 1)

23.6.3 Block Transfer Mode

Block transfer mode is different from normal smart card interface mode in the following respects.

- Even if a parity error is detected during reception, no error signal is output. Since the PER bit in SSR is set by error detection, clear the PER bit before receiving the parity bit of the next frame.
- During transmission, at least 1 etu is secured as a guard time from the end of the parity bit until the start of the next frame.
- Since the same data is not retransmitted during transmission, the TEND flag in SSR is set 11.5 etu after transmission start.
- In block transfer mode, the ERS flag in SSR indicates the error signal status as in normal smart card interface mode, but the flag is read as 0 because no error signal is transferred.

23.6.4 Receive Data Sampling Timing and Reception Margin

Only the internal clock generated by the on-chip baud rate generator can be used as a transfer clock in smart card interface mode.

In this mode, the SCI can operate on a base clock with a frequency of 32, 64, 372, 256, 93, 128, 186, or 512 times the bit rate according to the settings of the BCP2 bit in SCMR and the BCP[1:0] bits in SMR (the frequency is always 16 times the bit rate in normal asynchronous mode).

For data reception, the falling edge of the start bit is sampled with the base clock to perform internal synchronization. Receive data is sampled on the 16th, 32nd, 186th, 128th, 46th, 64th, 93rd, and 256th rising edges of the base clock so that it can be latched at the middle of each bit as shown in **Figure 23.33**. The reception margin here is determined by the following formula.

$$M = \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\ [\%]$$

M: Reception margin (%)
N: Ratio of bit rate to clock (N = 32, 64, 372, 256) D: Duty cycle of clock (D = 0 to 1.0)
L: Frame length (L = 10)
F: Absolute value of clock frequency deviation

Assuming values of F = 0, D = 0.5, and N = 372 in the above formula, the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 372)\} \times 100\ [\%] = 49.866\%$$

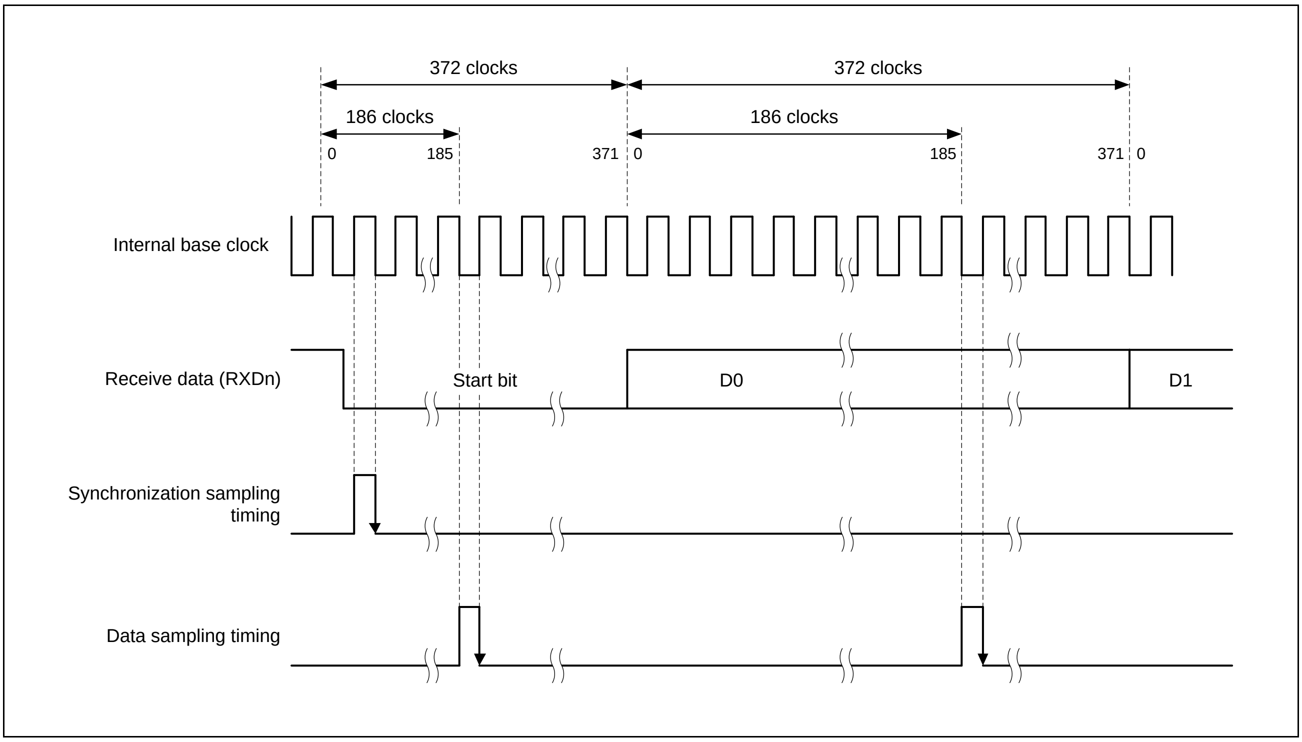


Figure 23.33 Receive Data Sampling Timing in Smart Card Interface Mode (When Clock Frequency is 372 Times the Bit Rate)

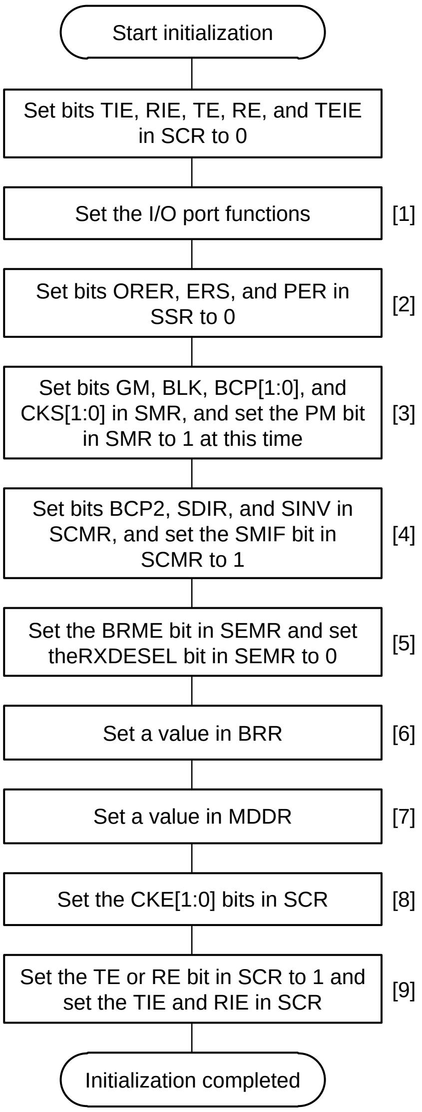
23.6.5 SCI Initialization (Smart Card Interface Mode)

Initialize the SCI following the example of flowchart shown in **Figure 23.34**.

Be sure to initialize the SCI before switching from transmission mode to reception mode and vice versa. Even if the RE bit is set to 0, the RDR register is not initialized.

To change reception mode to transmission mode, first check that reception has completed, and then initialize the SCI. At the end of initialization, set TE = 1 and RE = 0. Reception completion can be verified by reading the RXI request, ORER, or PER flag in SSR.

To change transmission mode to reception mode, first check that transmission has completed, and then initialize the SCI. At the end of initialization, set TE = 0 and RE = 1. Transmission completion can be verified by reading the TEND flag in SSR.



[1] Set the I/O ports so that the necessary pin functions can be enabled among the TXDn, RXDn, and SCKn pins.

[2] Set the ORER, ERS, and PER bits in SSR to 0.
After reading SSR, write 0 in the target bits.

[3] Set the transmission/reception format in SMR.

[4] Set the transmission/reception format in SCMR.
The TXDn and RXDn pins enter the high-impedance state.

[5] Set the BRME bit in SEMR and set the RXDESEL bit in SEMR to 0.

[6] Write the value corresponding to the bit rate in BRR.

[7] Write the value obtained by correcting a bit rate error in MDDR. This step is not necessary if the bit rate modulation function is not used.

[8] Set the CKE[1:0] bits in SCR.
When the CKE[0] bit is set to 1, the clock is output from the SCKn pin.

[9] Set the TE or RE bit in SCR to 1 and set the TIE, and RIE bits in SCR.
Do not simultaneously set the TE and RE bits to 1 if self-diagnosis is not used.

Figure 23.34 Example of SCI Initialization Flowchart (Smart Card Interface Mode)

23.6.6 Serial Data Transmission (Except in Block Transfer Mode)

Serial data transmission in smart card interface mode (except in block transfer mode), in that an error signal is sampled and data can be retransmitted, is different from that in non-smart card interface mode. **Figure 23.35** shows the data retransfer operation during transmission.

1. When an error signal from the receiver end is sampled after one-frame data has been transmitted, the ERS flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the ERS flag to 0 before the next parity bit is sampled.
2. For a frame in which an error signal is received, the TEND flag in SSR is not set. Data is retransferred from TDR to TSR allowing automatic data retransmission.
3. If no error signal is returned from the receiver, the ERS flag is not set to 1.
4. In this case, the SCI judges that transmission of one-frame data (including retransfer) has been completed, and the TEND flag is set. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Writing transmit data to TDR starts transmission of the next data.

Figure 23.37 shows a sample flowchart of serial transmission. All the processing steps are automatically performed using a TXI interrupt request to activate the DMAC.

When the TEND flag in SSR is set to 1 in transmission, if the TIE bit in SCR is 1, a TXI interrupt request is generated. The DMAC is activated by a TXI interrupt request if the TXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of transmit data. The TEND flag is automatically set to 0 when the DMAC transfers the data.

If an error occurs, the SCI automatically retransmits the same data. During this retransmission, the TEND flag is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, since the ERS flag is not automatically cleared, set the RIE bit to 1 beforehand to enable an ERI interrupt request to be generated at error occurrence, and clear the ERS flag to 0. When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings.

For DMAC settings, refer to **Section 14, Direct Memory Access Controller**.

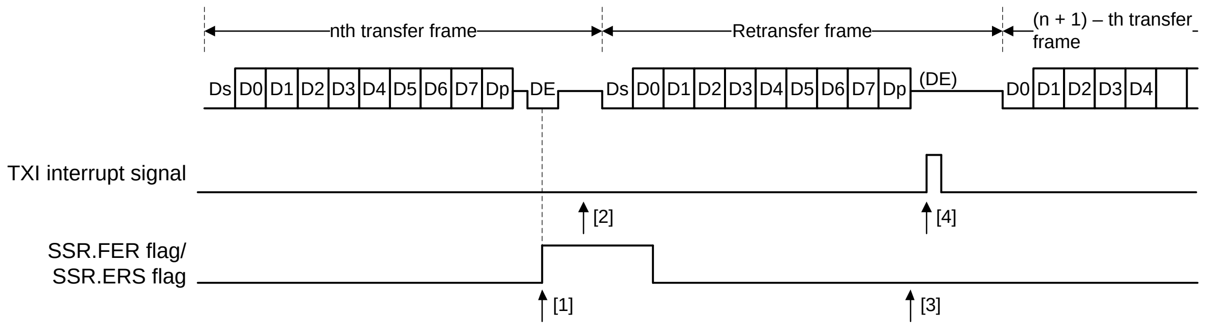
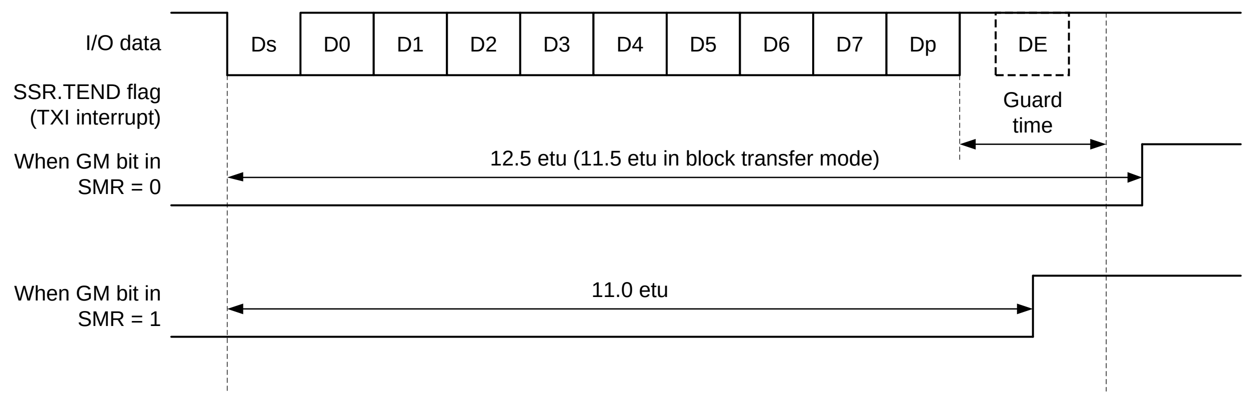


Figure 23.35 Data Retransfer Operation in SCI Transmission Mode

Note that the SSR.TEND flag is set in different timings depending on the GM bit setting in SMR. **Figure 23.36** shows the TEND flag generation timing.



Note: Ds: Start bit
D0 to D7: Data bits
Dp: Parity bit
DE: Error signal

Figure 23.36 SSR.TEND Flag Generation Timing during Transmission

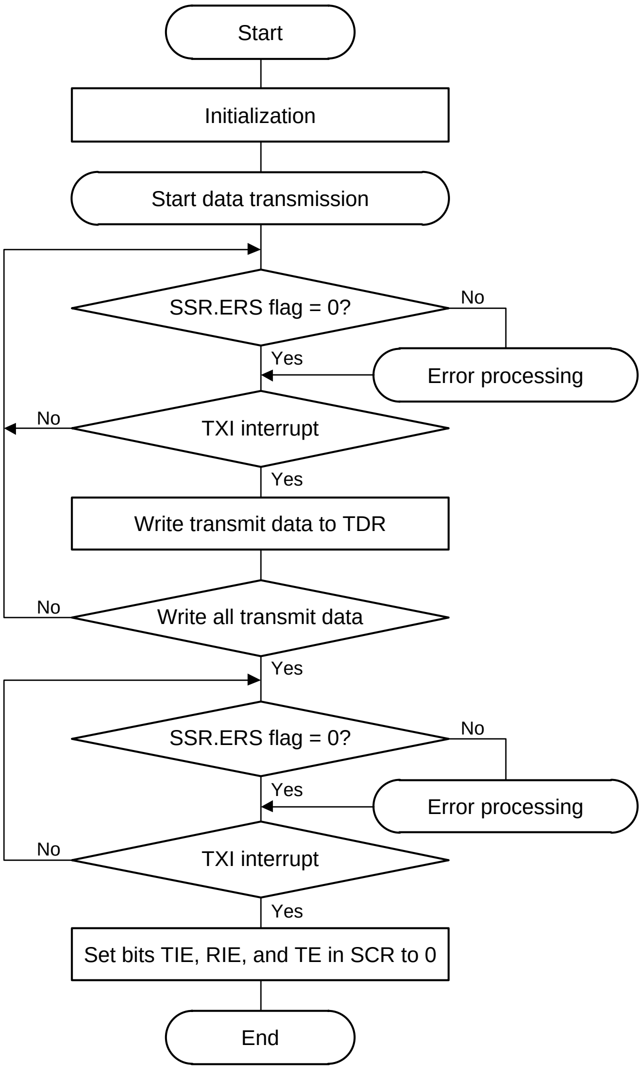


Figure 23.37 Sample Smart Card Interface Transmission Flowchart

23.6.7 Serial Data Reception (Except in Block Transfer Mode)

Serial data reception in smart card interface mode is similar to that in non-smart card interface mode. **Figure 23.38** shows the data retransfer operation in reception mode.

1. If a parity error is detected in receive data, the PER flag in SSR is set to 1. When the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the PER flag to 0 before the next parity bit is sampled.
2. For a frame in which a parity error is detected, no RXI interrupt is generated.
3. When no parity error is detected, the PER flag in SSR is not set to 1.
4. In this case, data is determined to have been received successfully. When the RIE bit in SCR is 1, an RXI interrupt request is generated.

Figure 23.39 shows a sample flowchart for serial data reception. All the processing steps are automatically performed using an RXI interrupt request to activate the DMAC.

In reception, setting the RIE bit to 1 allows an RXI interrupt request to be generated. The DMAC is activated by an RXI interrupt request if the RXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of receive data.

If an error occurs during reception and either the ORER or PER flag in SSR is set to 1, a receive error interrupt (ERI) request is generated. Clear the error flag after the error occurrence. If an error occurs, the DMAC is not activated and receive data is skipped. Therefore, the number of bytes of receive data specified in the DMAC is transferred.

Even if a parity error occurs and the PER flag is set to 1 during reception, receive data is transferred to RDR, thus allowing the data to be read.

When a reception is forcibly terminated by setting the RE bit in SCR to 0 during operation, read the RDR register because the received data which has not yet been read may be left in RDR.

NOTE

For operations in block transfer mode, refer to **Section 23.3, Operation in Asynchronous Mode**.

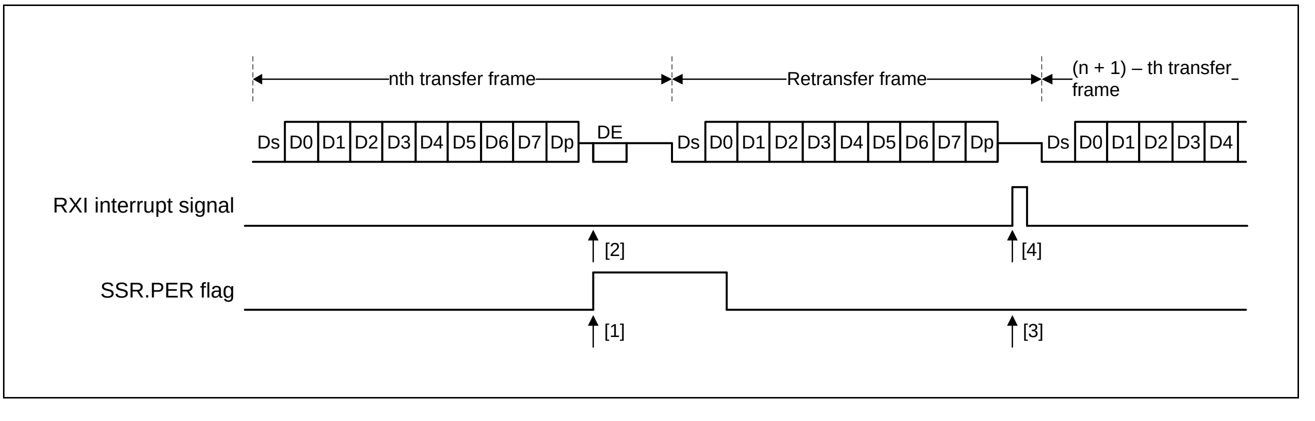


Figure 23.38 Data Retransfer Operation in SCI Reception Mode (Data Retransfer Operation during Reception)

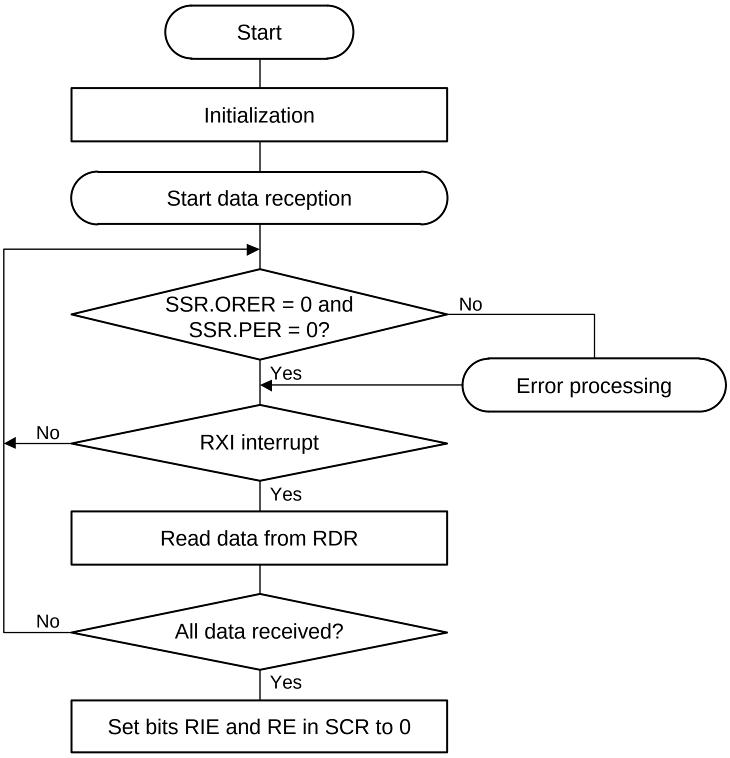


Figure 23.39 Sample Smart Card Interface Reception Flowchart

23.6.8 Clock Output Control

Clock output can be fixed using the CKE[1:0] bits in SCR when the GM bit in SMR is 1. Specifically, the minimum width of a clock pulse can be specified.

Figure 23.40 shows an example of clock output fixing timing when the CKE[0] bit is controlled with GM = 1 and CKE[1] = 0.

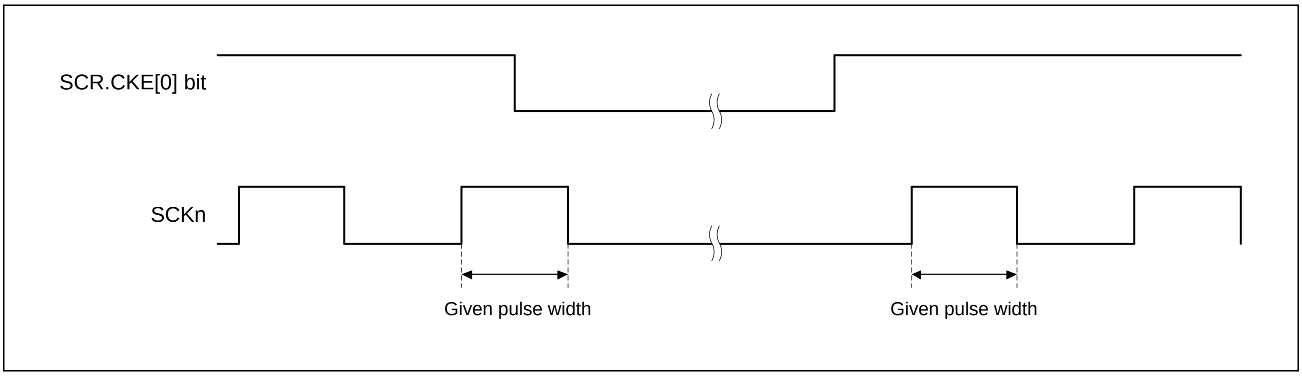


Figure 23.40 Clock Output Fixing Timing

At power-on, use the following procedure to secure the appropriate clock duty cycle.

(1) At Power-On

To secure the appropriate clock duty cycle simultaneously with power-on, use the following procedure.

1. Initially, port input is enabled in the high-impedance state. To fix the potential level, use a pull-up or pull-down resistor.
2. Fix the SCKn pin to the specified output by setting the SCR.CKE[1] bit and I/O port functions.
3. Set SMR and SCMR to enable smart card interface mode.
4. Set the SCR.CKE[0] bit to 1 to start clock output.

23.7 Noise Cancellation Function

Figure 23.41 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of two stages of flip-flop circuits and a match-detection circuit. When the level on the pin matches in three consecutive samples taken at the set sampling interval, the matching level continues to be conveyed internally until the level on the pin again matches in three consecutive samples.

In asynchronous mode, the noise cancellation function can be applied on the RXDn input signal. The period of the base clock (1/16th of a bit-period when SEMR.ABCS = 0 and 1/8th of a bit-period when SEMR.ABCS = 1) is the sampling interval.

If the base clock is stopped with the noise filter enabled and then the clock input is started again, the noise filter operation resumes from where the clock was stopped. If SCR.TE and SCR.RE are set to 0 during base clock input, all of the noise filter flip-flop values are initialized to 1. Accordingly, if the input data is 1 when reception operation resumes, it is determined that a level match is detected and is conveyed to the internal signal. When the level being input corresponds to 0, the initial output of the noise filter is retained until the level matches in three consecutive samples.

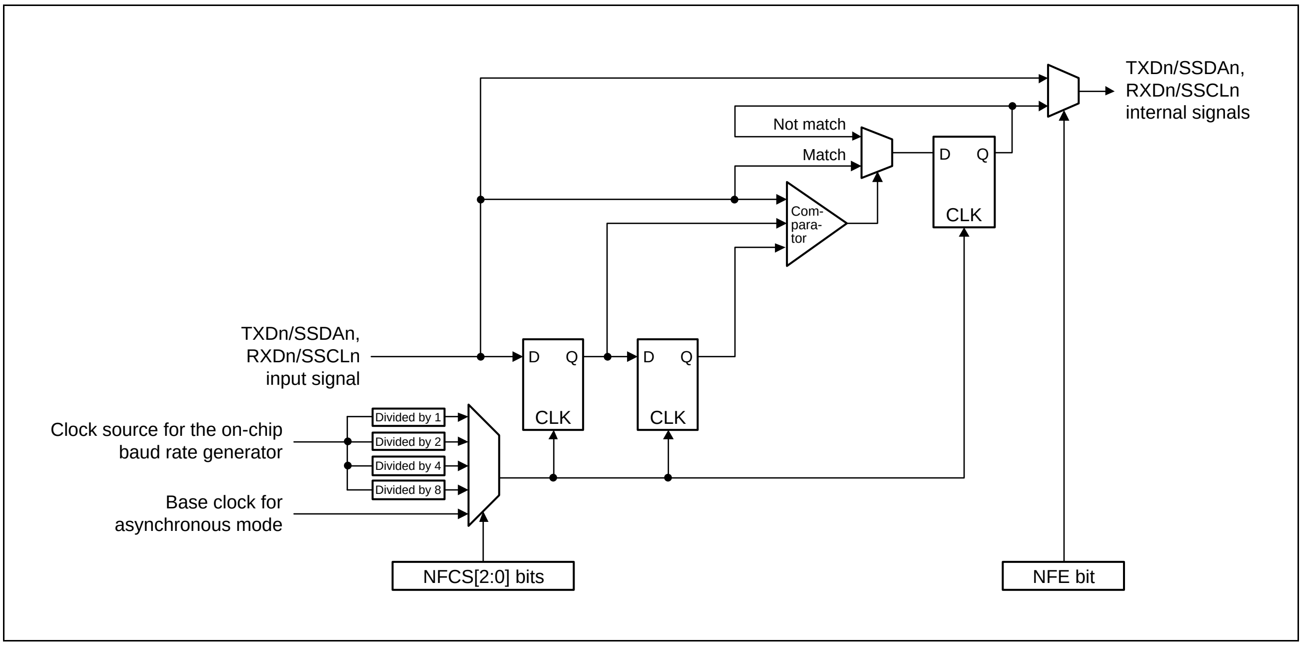


Figure 23.41 Block Diagram of Digital Noise Filter Circuit

23.8 Interrupt Sources

23.8.1 Buffer Operations for TXI and RXI Interrupts

If the conditions for a TXI and RXI interrupt are satisfied while the interrupt status flag in the interrupt controller is 1, the interrupt controller does not output the interrupt request but retains it internally (with a capacity for retention of one request per source).

When the value of the interrupt status flag in the interrupt controller becomes 0, the interrupt request retained within the interrupt controller is output. The internally retained interrupt request is automatically discarded once the actual interrupt is output. Clearing of the corresponding interrupt enable bit (the TIE or RIE bit in the SCR) can also be used to discard an internally retained interrupt request.

23.8.2 Interrupts in Asynchronous Mode and Clock Synchronous Mode

Table 23.20 lists interrupt sources in asynchronous mode and clock synchronous mode. A different interrupt vector is assigned to each interrupt source, and individual interrupt sources can be enabled or disabled with the enable bits in SCR. If the SCR.TIE bit is 1, a TXI interrupt request is generated when transmit data is transferred from the TDR or TDRL register*[1] to the TSR. A TXI interrupt request can also be generated by setting the SCR.TE bit to 1 after setting the SCR.TIE bit to 1 or by using a single instruction to set the SCR.TE and SCR.TIE bit to 1 at the same time. A TXI interrupt request can activate the DMAC to handle data transfer.

A TXI interrupt request is not generated by setting the SCR.TE bit to 1 while the setting of the SCR.TIE bit is 0 or by setting the SCR.TIE bit to 1 while the setting of the SCR.TE bit is 1.*[1]

When new data is not written by the time of transmission of the last bit of the current transmit data and the setting of the SCR.TEIE bit is 1, the SSR.TEND flag becomes 1 and a TEI interrupt request is generated. Furthermore, when the setting of the SCR.TE bit is 1, the SSR.TEND flag retains the value 1 until further transmit data are written to the TDR or TDRL register*[1], and setting the SCR.TEIE bit to 1 leads to the generation of a TEI interrupt request.

Writing data to the TDR or TDRL register*[1] leads to clearing of the SSR.TEND flag and, after a certain time, discarding of the TEI interrupt request.

If the SCR.RIE bit is 1, an RXI interrupt request is generated when received data is stored in the RDR. An RXI interrupt request can activate the DMAC to handle data transfer.

Setting of any from among the ORER, FER, and PER flags in the SSR to 1 while the SCR.RIE bit is 1 leads to the generation of an ERI interrupt request. An RXI interrupt request is not generated at this time. Clearing all three flags (ORER, FER, and PER) leads to discarding of the ERI interrupt request.

Note 1. In the case where asynchronous mode and 9-bit data length are selected

NOTE

To temporarily prohibit TXI interrupts at the time of transmission of the last of the data and so on when you wish a new round of transmission to start after handling of the transmission-completed interrupt, control prohibiting and permitting of the interrupt by using the interrupt request enable bit in the interrupt controller rather than using the SCR.TIE bit. This can prevent the suppression of TXI interrupt requests in the transfer of new data.

Table 23.20 Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	Priority
ERI	Receive error	ORER, FER, or PER	Not possible	High
RXI	Receive data full	—	Possible	↑
TXI	Transmit data empty	—	Possible	↑
TEI	Transmit end	TEND	Not possible	Low

23.8.3 Interrupts in Smart Card Interface Mode

Table 23.21 lists interrupt sources in smart card interface mode. A transmit end interrupt (TEI) request cannot be used in this mode.

Table 23.21 SCI Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	Priority
ERI	Receive error or error signal detection	ORER, PER, or ERS	Not possible	High
RXI	Receive data full	—	Possible	↑
TXI	Transmit data empty	TEND	Possible	Low

Data transmission/reception using the DMAC is also possible in smart card interface mode, similar to in the normal SCI mode. In transmission, when the TEND flag in SSR is set to 1, a TXI interrupt request is generated. This TXI interrupt request activates the DMAC allowing transfer of transmit data if the TXI request is specified beforehand as a source of DMAC activation. The TEND flag is automatically set to 0 when the DMAC transfers the data.

If an error occurs, the SCI automatically retransmits the same data. During the retransmission, the TEND flag is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, the ERS flag in SSR is not automatically cleared to 0 at error occurrence. Therefore, the ERS flag must be cleared by previously setting the RIE bit in SCR to 1 to enable an ERI interrupt request to be generated at error occurrence.

When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings. For DMAC settings, refer to **Section 14, Direct Memory Access Controller**.

In reception, an RXI interrupt request is generated when receive data is set to RDR. This RXI interrupt request activates the DMAC allowing transfer of receive data if the RXI request is specified beforehand as a source of DMAC activation. If an error occurs, the error flag is set. Therefore, the DMAC is not activated and an ERI interrupt request is issued to the CPU instead; the error flag must be cleared.

23.9 Usage Notes

23.9.1 Break Detection and Processing

When a framing error is detected, a break can be detected by reading the RXDn pin value directly. In a break, the input from the RXDn pin becomes all 0s, and so the FER flag in SSR is set to 1 (framing error has occurred), and the PER flag in SSR may also be set to 1 (parity error has occurred). The SCI continues the receive operation even after a break is received. Therefore, note that even if the FER flag is set to 0 (no framing error occurred), it will be set to 1 again. When the SEMR.RXDESEL bit is 1, the SCI sets the SSR.FER flag to 1 and stops receiving operation until a start bit of the next data frame is detected. If the SSR.FER flag is set to 0 at this time, the SSR.FER flag retains 0 during the break.

When the RXDn pin is set to 1 and the break ends, detecting the beginning of the start bit at the first falling edge of the RXDn pin allows the SCI to start the receiving operation.

23.9.2 Mark State and Production of Breaks

When the SCR.TE bit is 0 (serial transmission is disabled), setting the I/O port function makes selection of the level and direction (input or output) of the TXDn pin possible. If this is done, the TXDn pin can be placed in the mark state to send a break at the time of data transmission. Until the SCR.TE bit is set to 1 (serial transmission is enabled), the I/O port function is used to set the TXDn pin to output 1 and set the pin mode to a general I/O port pin, and thus place the communication line in the mark state (state of having the value 1). On the other hand, to output a break at the time of data transmission, set the TXDn pin to output 0 and make the pin mode settings for a general I/O port pin. When the SCR.TE bit is set to 0, the transmitter is initialized regardless of the current state of transmission.

23.9.3 Receive Error Flags and Transmit Operations (Clock Synchronous Mode)

Transmission cannot be started when a receive error flag (ORER) in SSR is set to 1, even if data is written to TDR. Be sure to set the receive error flags to 0 before starting transmission. Note also that the receive error flags cannot be set to 0 even if the RE bit in SCR is set to 0 (serial reception is disabled).

23.9.4 Writing Data to TDR

Data can be written to TDR, TDRH, and TDRL. However, if new data is written to TDR, TDRH, and TDRL when transmit data is remaining in TDR, TDRH, and TDRL, the previous data in TDR, TDRH, and TDRL is lost because it has not been transferred to TSR yet. Be sure to write transmit data to TDR, TDRH, and TDRL in the TXI interrupt request handling routine.

23.9.5 Restrictions on Clock Synchronous Transmission (Clock Synchronous Mode)

When the external clock source is used as a synchronization clock, the following restrictions apply.

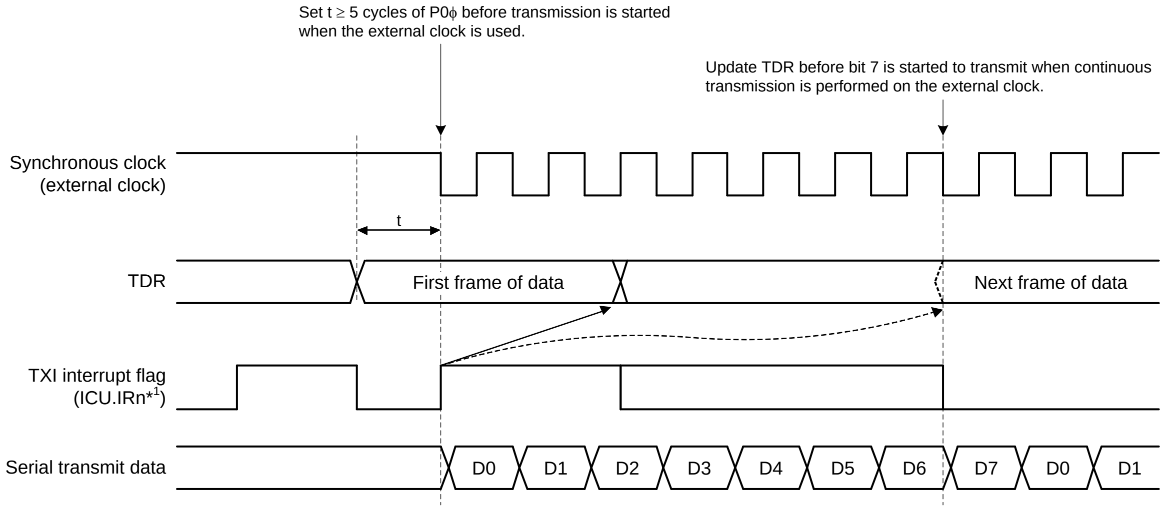
(1) Start of transmission

Update TDR by the CPU or DMAC and wait for at least five P0ϕ cycles before allowing the transmit clock to be input (see **Figure 23.42**).

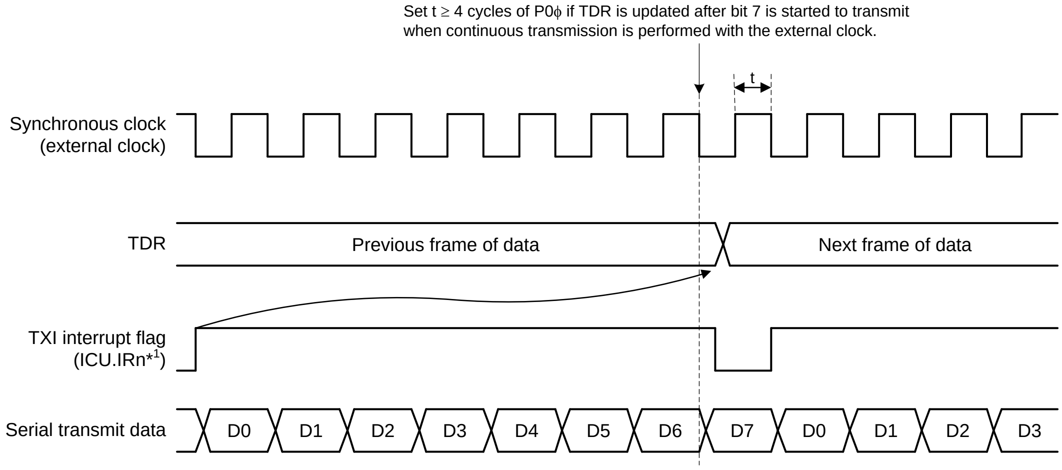
(2) Continuous transmission

(a) Write the next transmit data to TDR or TDRL before the falling edge of the transmit clock (bit 7) (see **Figure 23.42**).

(b) When updating TDR after bit 7 has started to transmit, update TDR while the synchronization clock is in the low-level period, and set the high-level width of the transmit clock (bit 7) to four P0ϕ cycles or longer (see **Figure 23.42**).



(a) Start of transmission and Continuous transmission (1)



(b) Continuous transmission (2)

Note 1. Refer to **Section 8, Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 23.42 Restrictions on Use of External Clock in Clock Synchronous Transmission

23.9.6 Restrictions on Using DMAC

When using the DMAC or DTC to read RDR, RDRH, and RDRL, be sure to set the receive data full interrupt (RXI) as the activation source of the relevant SCI.

23.9.7 Notes on Starting Transfer

At the point where transfer starts when the interrupt status flag (IRn.IR bit) in the interrupt controller is 1, follow the procedure below to clear interrupt requests before permitting operations (by setting the SCR.TE or SCR.RE bit to 1). For details on the interrupt status flag, refer to **Section 8, Interrupt Controller**.

- Confirm that transfer has stopped (the setting of the SCR.TE or SCR.RE bits is 0).
- Set the corresponding interrupt enable bit (SCR.TIE or SCR.RIE) to 0.
- Read the corresponding interrupt enable bit (SCR.TIE or SCR.RIE bit) to check that it has become 0.
- Set the interrupt status flag (IRn.IR bit) in the interrupt controller to 0.

23.9.8 External Clock Input in Clock Synchronous Mode

In clock synchronous mode, the external clock SCKn must be input as follows:

High-pulse period, low-pulse period = 2 P0ϕ cycles or more, period = 6 P0ϕ cycles or more

23.9.9 Note on Transmit Enable Bit (TE bit)

When setting the SCR.TE bit to 0 (serial transmission is disabled) while the pin function is “TXDn”, output of the pin becomes high impedance.

Prevent the TXDn line from becoming high impedance by any of the following ways:

(1) Connect a pull-up resistor to the TXDn line.

(2) Change the pin function to “general-purpose I/O port, output” before setting the SCR.TE bit to 0. Set the SCR.TE bit to 1 before changing the pin function to “TXDn”.

23.10 IrDA Communications

The channel 0 serial communications interface (SCI) is capable of working with the Infrared Data Association (IrDA) module to handle transfer through IrDA communications in conformance with version 1.0 of the IrDA protocol.

Using the IRE bit in IRCR to enable the IrDA function leads to encoding and decoding of the SCI0_TXD and SCI0_RXD signals of the channel 0 serial communications interface, respectively, to convert them to and from waveforms in conformance with version 1.0 of the IrDA protocol. Infrared transfer in conformance with version 1.0 of the IrDA protocol can be handled by connecting a transceiver for use in infrared communications.

In version 1.0 of the IrDA protocol, IrDA transfer starts at a transfer rate of 9600 bps. The transfer rate is changeable during transfer as required. The IrDA module itself does not have a function for automatically changing the transfer rate. Accordingly, change the transfer rate by changing that for the serial communications interface.

A block diagram and pin configuration for IrDA transfer are shown in **Figure 23.43** and **Table 23.22**, respectively.

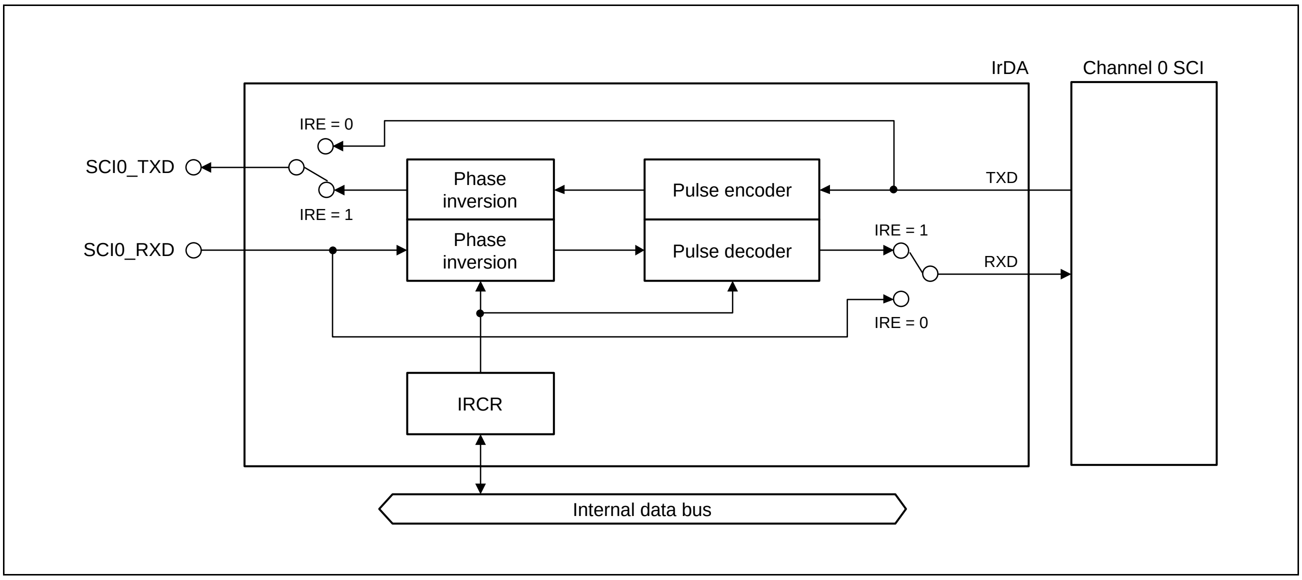


Figure 23.43 Block Diagram

Table 23.22 Pin Configuration

Pin Name	Symbol	I/O	Function
IrDA data transmission pin	SCI0_TXD	Output	IrDA data transmission output
IrDA data reception pin	SCI0_RXD	Input	IrDA data reception input

23.11 Description of the IrDA Register

Table 23.23 shows the register configuration.

BASE Address (Cortex-A55 Address Space): H’0_1004_CC00
BASE Address (Cortex-M33 Address Space Non-Secure): H’4004_CC00
BASE Address (Cortex-M33 Address Space Secure): H’5004_CC00

Table 23.23 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
IrDA Control Register	IRCR	R/W	H’00	H’0_1004_CC00	8

23.11.1 IrDA Control Register (IRCR)

IRCR sets the operation of the IrDA module.

Bit	b7	b6	b5	b4	b3	b2	b1	b0
	IRE	IRCKS[2:0]			IRTXINV	IRRXINV	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W								

Bit	Bit Name	Initial Value	R/W	Description
b1, b0	—		R	Reserved These bits are read as 0. The write value should be 0.
b2	IRRXINV		R/W	SCI_RXD0 Data Level Switching Sets inversion of the logic level of inputs on SCI0_RXD. 0: The input on SCI0_RXD is used without change as the received data. 1: The input on SCI0_RXD is inverted before it is treated as the received data.
b3	IRTXINV		R/W	SCI_TXD0 Data Level Switching Sets inversion of the logic level of outputs on SCI0_TXD. 0: Data for transmission is output without change to SCI0_TXD. The pulse-width setting in the IRCKS bits determines the period over which the pulses are at the high level. 1: Data for transmission is inverted before being output to SCI0_TXD. The pulse-width setting in the IRCKS bits determines the period over which the pulses are at the low level.
b6 to b4	IRCKS[2:0]		R/W	IrDA Clock Select Selects the pulse width for use in encoding output pulses from SCI0_TXD when the setting of the IRE bit is 1. 000: Bit rate × 3/16 Settings other than above are prohibited.
b7	IRE		R/W	IrDA Enable Sets the function for use with the SCI0_TXD and SCI0_RXD pins as normal serial or IrDA. 0: Disables the IrDA function. Data from the TXD pin of the channel 0 serial communications interface is output without change to SCI0_TXD. Data on SCI0_RXD is output without change to the RXD pin of the channel 0 serial communications interface. 1: Enables the IrDA function. Data from the TXD pin of the channel 0 serial communications interface is encoded before being output through SCI0_TXD. Data on SCI0_RXD is decoded before being output to the RXD pin of the channel 0 serial communications interface.

23.12 IrDA Operation

23.12.1 Flow of Settings for IrDA Operation

Set up IrDA operation by following the procedure listed below.

1. Make the general-purpose I/O port-pin settings.
2. Set the IRCR register.
3. Set the registers related to the serial communications interface.

23.12.2 Transmission

In transmission with the IrDA function enabled, serial data (UART frames of data) from the TXD pin of the serial communications interface are converted to IR frames (see **Figure 23.44**). If the IRTXINV bit is set to 0, when the value of the serial data is 0, a high-level pulse with the width equivalent to three sixteenths of the bit period is output to the SCI0_TXD pin by default. In the IrDA protocol, the specifiable high-level pulse widths are from a minimum of 1.41 µs to a maximum of (3/16 + 2.5%) × bit period or (3/16 × bit period) + 1.08 µs. On the other hand, if the value of the serial data is 1, no high-level pulse is output.

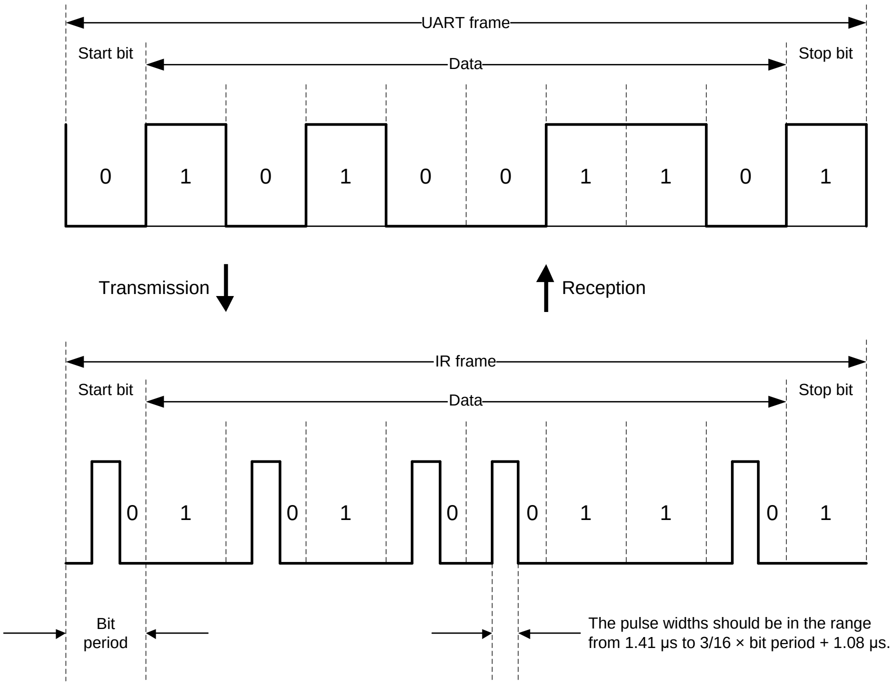


Figure 23.44 Example of the Operation in IrDA Transfer

23.12.3 Reception

In reception with the IrDA function enabled, IR frames of data from the SCI0_RXD pin are converted to serial data and output to the RXD pin of the serial communications interface. If the setting of the IRRXINV bit is 0, the value 0 is output when a high-level pulse is detected and the value 1 is output when no high-level pulse is detected during each bit period. Note that pulses with a width shorter than the minimum pulse width of 1.41 μs cannot be recognized.

23.12.4 Relationship between Bit Rate and High-Level Pulse Width

The relationship between the Bit Rate during transmission and the High-Level Pulse is shown in **Table 23.24**.

Table 23.24 Upper Row: Bit Rate (bps); Lower Row: Bit Period × 3/16 (μs)

2400	9600	19200	38400	57600	115200
78.13	19.53	9.77	4.88	3.26	1.63

23.13 Note on IrDA Usage

23.13.1 Minimum Pulse Width in Reception

Pulses with a width shorter than the minimum pulse width of 1.41 μs cannot be recognized

23.13.2 Base Clock in Asynchronous Mode for the Serial Communications Interface

The IrDA module works with the serial communications interface by receiving the base clock with a frequency 16 times the bit rate for communications from the interface. For the serial communications interface, either 16 or 8 cycles of the base clock are selectable as the bit period. On the other hand, the IrDA module is only capable of operation when the setting for the bit period of the serial communications interface is 16 cycles of the base clock.

24. Renesas Serial Peripheral Interface

This LSI circuit includes three independent Renesas serial peripheral interfaces.

This module is capable of full-duplex synchronous serial communication.

24.1.1 Features

This module has the following features.

SPI transfer functions

- Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (SPI clock) signals allows for serial communications through SPI operation (four-wire method).
- Capable of serial communications in master/slave mode
- Supports mode fault error detection (only in SPI slave mode)
- Supports overrun error detection (only in SPI slave mode)
- Switching of the polarity of the serial transfer clock
- Switching of the clock phase of serial transfer

Data format

- MSB-first/LSB-first selectable
- Transfer bit-length is selectable as 8, 16, or 32 bits.

Bit rate

- RSPCK can be divided by a maximum of 4096 in master mode
- RSPCK can be generated by dividing P0ϕ by the on-chip baud rate generator.
- An externally input clock can be used as a serial clock.

Buffer configuration

- 8 bytes for transmission and 32 bytes for reception

SSL control function

- One SSL signal for each channel
    - In master mode, outputs SSL signal.
    - In slave mode, inputs SSL signal.
- Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay)
    - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
- Controllable delay from RSPCK stoppage to SSL output negation (SSL negation delay)
    - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
- Controllable wait for next-access SSL output assertion (next-access delay)
    - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
- Function for changing SSL polarity

Control in master transfer

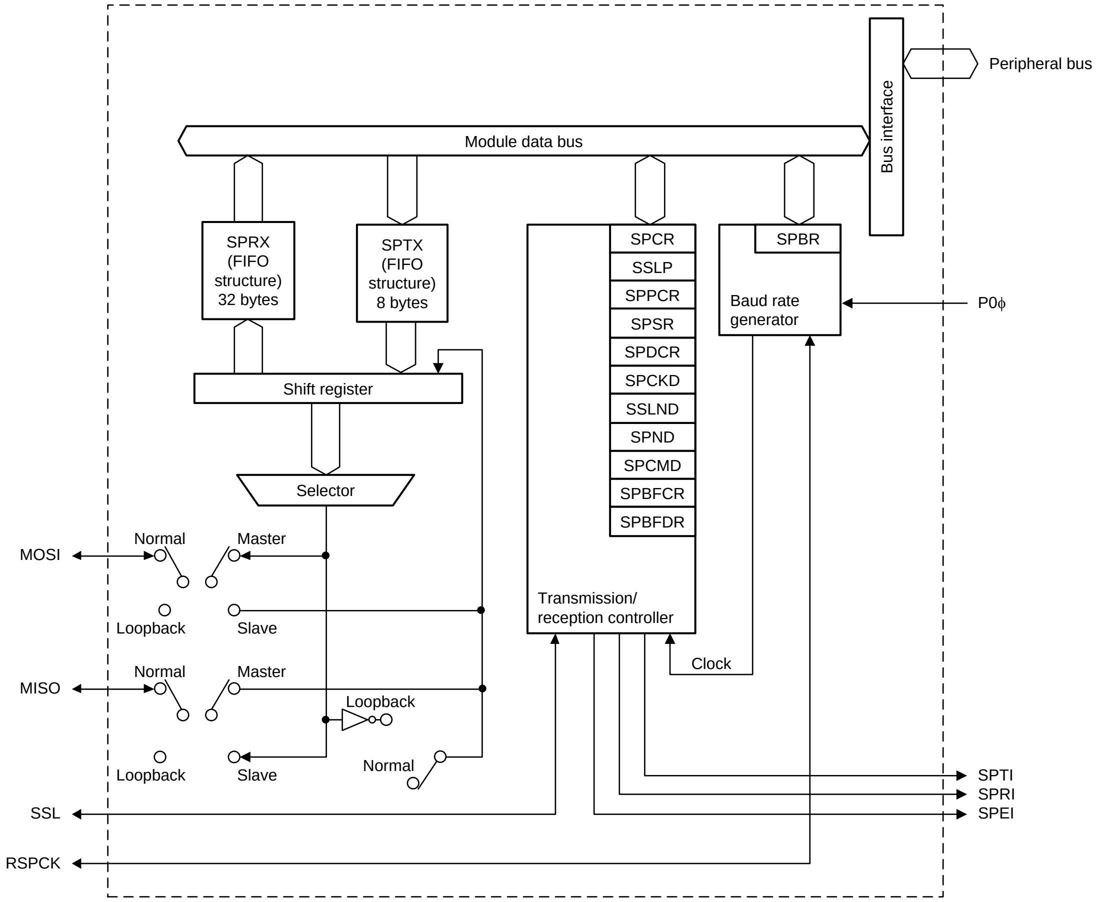
- A transfer of up to four commands can be executed sequentially in looped execution.
- For each command, the following can be set:
  - SSL signal value, bit rate, RSPCK polarity/phase, transfer data length, LSB/MSB first, burst, RSPCK delay, SSL negation delay, and next-access delay.
  - A transfer can be initiated by writing to the transmit buffer.
  - A transfer can be initiated by clearing the SPTEF bit.
  - MOSI signal value specifiable in SSL negation

Interrupt sources

- Maskable interrupt sources:
  - Receive interrupt (receive buffer full)
  - Transmit interrupt (transmit buffer empty)
  - Error interrupt (mode fault, overrun)

Others

- Provides loop back mode
- Provides a function for disabling (initializing) this module



[Legend]

SPCR: Control register

SSLP: Slave select polarity register

SPPCR: Pin control register

SPSR: Status register

SPSCR: Sequence control register

SPSSR: Sequence status register

SPDCR: Data control register

SPCKD: Cock delay register

SSLND: Slave select negate delay register

SPND: Next-access delay register

SPCMD: Command register

SPBR: Bit rate register

SPTX: Transmission buffer (Data register write side)

SPRX: Receive buffer (Data register read side)

SPBFCR: Buffer control register

SPBFDR: Buffer data count setting register

SPTI: Transmit interrupt

SPRI: Receive interrupt

SPEI: Error interrupt

Figure 24.1 Block Diagram (for One Channel)

24.2 Input/Output Pins

Table 24.1 shows the pin configuration. This module automatically switches the input/output direction of the SSL pin. SSL is set as an output in master mode and as an input in slave mode. Pins RSPCK, MOSI, and MISO are automatically set as inputs or outputs according to the setting of master or slave and the level input on SSL (see **Section 24.4.2, Pin Control**).

Table 24.1 Pin Configuration

Channel	Pin Name	Pin Name	I/O	Function
0	Clock pin	RSPI0_CK	I/O	Clock input/output
	Master transmit data pin	RSPI0_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI0_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI0_SSL	I/O	Slave selection
1	Clock pin	RSPI1_CK	I/O	Clock input/output
	Master transmit data pin	RSPI1_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI1_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI1_SSL	I/O	Slave selection
2	Clock pin	RSPI2_CK	I/O	Clock input/output
	Master transmit data pin	RSPI2_MOSI	I/O	Master transmit data
	Slave transmit data pin	RSPI2_MISO	I/O	Slave transmit data
	Slave select 0 pin	RSPI2_SSL	I/O	Slave selection

Note: In the description of the pins, the channel is omitted, and pin names are described as RSPCK, MOSI, MISO, and SSL.

24.3 Register Descriptions

Table 24.2 shows the register configuration. The address of the RSPI register is represented by the offset address from the base address. RSPI base address is as follows:

Base address: H’0_1004_0000 (Overall Address Space)
Base address: H’5004_0000 (Cortex-M33 Address Space Secure)
Base address: H’4004_0000 (Cortex-M33 Address Space Non-Secure)

Table 24.2 Register Configuration (1/2)

Channel	Register Name	Abbreviation*[1]	R/W	Initial Value	Offset Address	Access Size
0	Control register_0	SPCR_0	R/W	H’00	H’AC00	8
	Slave select polarity register_0	SSLP_0	R/W	H’00	H’AC01	8
	Pin control register_0	SPPCR_0	R/W	H’00	H’AC02	8
	Status register_0	SPSR_0	R/(W)*[2]	H’60	H’AC03	8
	Data register_0	SPDR_0	R/W	Undefined	H’AC04	8, 16, 32
	Sequence control register_0	SPSCR_0	R/W	H’00	H’AC08	8
	Sequence status register_0	SPSSR_0	R	H’00	H’AC09	8
	Bit rate register_0	SPBR_0	R/W	H’FF	H’AC0A	8
	Data control register_0	SPDCR_0	R/W	H’20	H’AC0B	8
	Clock delay register_0	SPCKD_0	R/W	H’00	H’AC0C	8
	Slave select negation delay register_0	SSLND_0	R/W	H’00	H’AC0D	8
	Next-access delay register_0	SPND_0	R/W	H’00	H’AC0E	8
	Command register0_0	SPCMD0_0	R/W	H’070D	H’AC10	16
	Command register1_0	SPCMD1_0	R/W	H’070D	H’AC12	16
	Command register2_0	SPCMD2_0	R/W	H’070D	H’AC14	16
	Command register3_0	SPCMD3_0	R/W	H’070D	H’AC16	16
	Buffer control register_0	SPBFCR_0	R/W	H’00	H’AC20	8
	Buffer data count setting register_0	SPBFDR_0	R	H’0000	H’AC22	16
1	Control register_1	SPCR_1	R/W	H’00	H’B000	8
	Slave select polarity register_1	SSLP_1	R/W	H’00	H’B001	8
	Pin control register_1	SPPCR_1	R/W	H’00	H’B002	8
	Status register_1	SPSR_1	R/(W)*[2]	H’60	H’B003	8
	Data register_1	SPDR_1	R/W	Undefined	H’B004	8, 16, 32
	Sequence control register_1	SPSCR_1	R/W	H’00	H’B008	8
	Sequence status register_1	SPSSR_1	R	H’00	H’B009	8
	Bit rate register_1	SPBR_1	R/W	H’FF	H’B00A	8
	Data control register_1	SPDCR_1	R/W	H’20	H’B00B	8
	Clock delay register_1	SPCKD_1	R/W	H’00	H’B00C	8
	Slave select negation delay register_1	SSLND_1	R/W	H’00	H’B00D	8
	Next-access delay register_1	SPND_1	R/W	H’00	H’B00E	8
	Command register0_1	SPCMD0_1	R/W	H’070D	H’B010	16
	Command register1_1	SPCMD1_1	R/W	H’070D	H’B012	16
	Command register2_1	SPCMD2_1	R/W	H’070D	H’B014	16
	Command register3_1	SPCMD3_1	R/W	H’070D	H’B016	16
	Buffer control register_1	SPBFCR_1	R/W	H’00	H’B020	8
	Buffer data count setting register_1	SPBFDR_1	R	H’0000	H’B022	16

Table 24.2 Register Configuration (2/2)

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Offset Address	Access Size
2	Control register_2	SPCR_2	R/W	H'00	H'B400	8
	Slave select polarity register_2	SSLP_2	R/W	H'00	H'B401	8
	Pin control register_2	SPPCR_2	R/W	H'00	H'B402	8
	Status register_2	SPSR_2	R/(W)*2	H'60	H'B403	8
	Data register_2	SPDR_2	R/W	Undefined	H'B404	8, 16, 32
	Sequence control register_2	SPSCR_2	R/W	H'00	H'B408	8
	Sequence status register_2	SPSSR_2	R	H'00	H'B409	8
	Bit rate register_2	SPBR_2	R/W	H'FF	H'B40A	8
	Data control register_2	SPDCR_2	R/W	H'20	H'B40B	8
	Clock delay register_2	SPCKD_2	R/W	H'00	H'B40C	8
	Slave select negation delay register_2	SSLND_2	R/W	H'00	H'B40D	8
	Next-access delay register_2	SPND_2	R/W	H'00	H'B40E	8
	Command register0_2	SPCMD0_2	R/W	H'070D	H'B410	16
	Command register1_2	SPCMD1_2	R/W	H'070D	H'B412	16
	Command register2_2	SPCMD2_2	R/W	H'070D	H'B414	16
	Command register3_2	SPCMD3_2	R/W	H'070D	H'B416	16
	Buffer control register_2	SPBFCR_2	R/W	H'00	H'B420	8
	Buffer data count setting register_2	SPBFDR_2	R	H'0000	H'B422	16

Note 1. In the description of the register names, the channel is omitted.

Note 2. Only 0 can be written to clear the flag.

24.3.1 Control Register (SPCR)

SPCR sets the operating mode. If the MSTR and MODFEN bits are changed while the function of this module is enabled by setting the SPE bit to 1, subsequent operations cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	—	—
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	SPRIE	0	R/W	Receive Interrupt Enable Enables or disables generation of receive interrupt requests (SPRI) when the number of receive data units in the receive buffer (SPRX) is equal to or greater than the specified receive buffer data triggering number and the SPRF flag in SPSR is set to 1. 0: Disables the generation of receive interrupt requests. 1: Enables the generation of receive interrupt requests.
6	SPE	0	R/W	Function Enable Setting this bit to 1 enables the module function. When the MODF bit in the status register (SPSR) is 1, the SPE bit cannot be set to 1 (see **Section 24.4.6, Error Detection**). Setting the SPE bit to 0 disables the module function, and initializes a part of the module function (see **Section 24.4.7, Initialization**). 0: Disables the module function. 1: Enables the module function.
5	SPTIE	0	R/W	Transmit Interrupt Enable Enables or disables generation of transmit interrupt requests (SPTI) when the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the specified transmit buffer data triggering number and the SPTEF flag in SPSR is set to 1. 0: Disables the generation of transmit interrupt requests. 1: Enables the generation of transmit interrupt requests.
4	SPEIE	0	R/W	Error Interrupt Enable Enables or disables the generation of error interrupt requests when this module detects a mode fault error and sets the MODF bit in the status register (SPSR) to 1, or when this module detects an overrun error and sets the OVRF bit in SPSR to 1 (see **Section 24.4.6, Error Detection**). 0: Disables the generation of error interrupt requests. 1: Enables the generation of error interrupt requests. *Note*: This bit is valid only in SPI slave mode.
3	MSTR	0	R/W	Master/Slave Mode Select Selects master/slave mode. According to MSTR bit settings, this module determines the direction of the RSPCK, MOSI, MISO, and SSL pins. 0: Slave mode 1: Master mode
2	MODFEN	0	R/W	Mode Fault Error Detection Enable Enables or disables the detection of a mode fault error (see **Section 24.4.6, Error Detection**). 0: Disables the detection of a mode fault error. 1: Enables the detection of a mode fault error. *Note*: This bit is valid only in SPI slave mode. When master mode is specified with the MSTR bit, this bit should always be cleared to 0.
1, 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

24.3.2 Slave Select Polarity Register (SSLP)

SSLP sets the polarity of the SSL signal. If the contents of SSL0P are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSL0P
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SSL0P	0	R/W	SSL Signal Polarity Setting Sets the polarity of the SSL signal. The value of SSL0P indicates the active polarity of the SSL signal. 0: SSL signal 0-active 1: SSL signal 1-active

24.3.3 Pin Control Register (SPPCR)

SPPCR sets the modes of the pins. If the contents of this register are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	—	—	SPLP
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5	MOIFE	0	R/W	MOSI Idle Value Fixing Enable Fixes the MOSI output value when this module in master mode is in an SSL negation period (including the SSL retention period during a burst transfer). When MOIFE is 0, this module outputs the last output value from the previous serial transfer during the SSL negation period to the MOSI pin. (The value is undefined when CPHA is 0). When MOIFE is 1, this module outputs the fixed value set in the MOIFV bit to the MOSI pin. 0: MOSI output value equals the last output value from previous transfer. (The value is undefined when CPHA is 0). 1: MOSI output value equals the value set in the MOIFV bit.
4	MOIFV	0	R/W	MOSI Idle Fixed Value If the MOIFE bit is 1 in master mode, this module, according to MOIFV bit settings, determines the MOSI signal value during the SSL negation period (including the SSL retention period during a burst transfer). 0: MOSI Idle fixed value equals 0. 1: MOSI Idle fixed value equals 1.
3 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SPLP	0	R/W	Loopback When the SPLP bit is set to 1, this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects (reverses) the input path and the output path for the shift register. 0: Normal mode 1: Loopback mode

24.3.4 Status Register (SPSR)

SPSR indicates the operating status.

Bit	7	6	5	4	3	2	1	0
	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
Initial Value	0	1	1	0	0	0	0	0
R/W	R	R	R	R	R	R/(W)*1	R	R/(W)*1

Note 1. Only 0 can be written to clear the flag after reading 1.

Bit	Bit Name	Initial Value	R/W	Description
7	SPRF	0	R	Receive Buffer Full Flag Indicates that the number of receive data units in the receive buffer (SPRX) is equal to or greater than the receive buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of receive data units in the receive buffer is less than the receive buffer data triggering number. 1: The number of receive data units in the receive buffer is equal to or greater than the receive buffer data triggering number. [Clearing conditions] ● The receive buffer data is read until the number of data units in the receive buffer becomes less than the specified receive buffer data triggering number. ● Receive buffer data reset is enabled. ● Power-on reset [Setting condition] ● The number of data units in the receive buffer is equal to or greater than the specified receive buffer data triggering number.
6	TEND	1	R	Transmit End This bit is set to 1 when transmission is completed, and this bit is 0 when transmission is not completed. [Clearing condition] ● When transmit data are transferred from the transmit register to the shift register. [Setting condition] ● When the number of data units in the transmit buffer (SPTX) is zero when a serial transfer is completed. *Note*: This bit is valid only in SPI master mode.
5	SPTEF	1	R	Transmit Buffer Empty Flag Indicates that the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the transmit buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of transmit data units in the transmit buffer is equal to or greater than the specified transmit buffer data triggering number. 1: The number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. [Clearing condition] ● When data is written to the transmit buffer until the number of transmit data units in the transmit buffer exceeds the specified transmit buffer data triggering number. [Setting conditions] ● When the number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. ● When transmit buffer data reset is enabled. ● Power-on reset
4, 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
2	MODF	0	R/(W)*1	Mode Fault Error Flag Indicates the occurrence of a mode fault error. If the MODFEN bit is set to 1 when this module is in slave mode and the SSL pin is negated before the RSPCK cycle necessary for data transfer ends, this module detects a mode fault error. The active level of the SSL signal is determined by the SSL0P bit in the slave select polarity register (SSLP). [Clearing conditions] ● SPSR is read when the MODF bit is 1, and then 0 is written to the MODF bit. ● Power-on reset 0: No mode fault error occurred 1: A mode fault error occurred *Note*: This bit is valid only in SPI slave mode.
1	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	OVRF	0	R/(W)*1	Overrun Error Flag Indicates the occurrence of an overrun error. If a serial transfer ends when there is not enough space for receiving the specified length of data in the receive buffer (SPRX), this module detects an overrun error, and sets the OVRF bit to 1. [Clearing conditions] ● SPSR is read when the OVRF bit is 1, and then 0 is written to the OVRF bit. ● Power-on reset 0: No overrun error occurred 1: An overrun error occurred *Note*: This bit is valid only in SPI slave mode.

Note 1. Only 0 can be written to clear the flag after reading 1.

24.3.5 Data Register (SPDR)

SPDR is a buffer that holds data for transmission and reception.

The transmit buffer (SPTX) and receive buffer (SPRX) are independent and are mapped to SPDR.

SPDR should be read or written to in byte, word, or longword units according to the access width specification bit (SPLW) in the data control register (SPDCR).

The bit length to be used is determined by the data length specification bits (SPB3 to SPB0) in the command register (SPCMD).

The access width set by SPDCR must agree with the data length set by SPCMD.

When data is written to SPDR, the data will be written to the transmit buffer from SPDR if the transmit buffer has a space equal to or more than the SPDR access width. If there is not enough space, data will not be written to the transmit buffer. Even if an attempt is made to write data to the buffer, the data is ignored.

When data is read from SPDR, receive data in the receive buffer will be read. If SPDR is read when there is no receive data in the receive buffer, the read value is undefined.

When SPDR is written to with the longword-, word-, or byte-access width, the transmit data should be written to address 0 irrespective of the access width. If data is written to the other addresses, the data is not guaranteed.

When SPDR is read with the longword-, word-, or byte-access width, the receive data should be read from address 0. If data is read from the other addresses, the data is not guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
Initial Value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
Initial Value	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

24.3.6 Sequence Control Register (SPSCR)

SPSCR sets the sequence control method when this module operates in master mode. If the contents of SPSCR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSLN1	SPSLN0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1	SPSLN1	0	R/W	Sequence Length Specification These bits specify a sequence length when this module in master mode performs sequential operations. This module in master mode changes command registers 0 to 3 (SPCMD0 to SPCMD3) to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN1 and SPSLN0 bits. The relationship among the setting of bits SPSLN1 and SPSLN0, sequence length, and SPCMD0 to SPCMD3 referenced by this module is shown below. In slave mode, SPCMD0 is always referenced. Sequence Length / Referenced SPCMD # 00: 1 0 → 0 → … 01: 2 0 → 1 → 0 → … 10: 3 0 → 1 → 2 → 0 → … 11: 4 0 → 1 → 2 → 3 → 0 → …
0	SPSLN0	0	R/W	

24.3.7 Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status when this module operates in master mode.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPCP1	SPCP0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1	SPCP1	0	R	Command Pointer During sequence control, these bits indicate one of the command registers 0 to 3 (SPCMD0 to SPCMD3) that is currently pointed to by the pointer. The relationship between the setting of SPCP1 and SPCP0 and SPCMD0 to SPCMD3 is shown below. For the sequence control, see **Section 24.4.8(1)(c), Sequence Control**. 00: SPCMD0 01: SPCMD1 10: SPCMD2 11: SPCMD3
0	SPCP0	0	R	

24.3.8 Bit Rate Register (SPBR)

SPBR sets the bit rate in master mode. If the contents of SPBR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When this module is used in slave mode, the bit rate depends on the bit rate of the input clock regardless of the settings of SPBR and BRDV.

The bit rate is determined by combinations of SPBR settings and the bit settings in the BRDV1 and BRDV0 bits in the command registers (SPCMD0 to SPCMD3). The equation for calculating the bit rate is given below. In the equation, n denotes an SPBR setting (0, 1, 2, …, 255), and N denotes a BRDV1 and BRDV0 bit setting (0, 1, 2, 3).

$$\text{Bit rate} = \frac{\mathrm{f}\,(\mathrm{P0}\phi)}{2 \times (\mathrm{n} + 1) \times 2^{\mathrm{N}}}$$

Table 24.3 shows examples of the relationship between the SPBR register and BRDV1 and BRDV0 bit settings.

Table 24.3 Relationship between SPBR and BRDV1 and BRDV0 Settings

SPBR (n)	BRDV[1:0] (N)	Division Ratio	Bit Rate*1
			P0ϕ = 100 MHz
0	0	2	Not Available
1	0	4	25.00 Mbps
2	0	6	16.67 Mbps
3	0	8	12.50 Mbps
4	0	10	10.00 Mbps
5	0	12	8.33 Mbps
5	1	24	4.17 Mbps
5	2	48	2.08 Mbps
5	3	96	1.04 Mbps
255	3	4096	24.41 Kbps

Note 1. Decide the bit rate to be actually used in the system considering timing specifications.

24.3.9 Data Control Register (SPDCR)

SPDCR selects the width to access SPDR from longword-, word-, and byte-width, and enables or disables dummy data transmission for the master mode operation.

If the contents of SPDCR are changed while bit TEND in the status register (SPSR) indicates that transmission is not completed, the subsequent operation cannot be guaranteed.

Bit	7	6	5	4	3	2	1	0
	TXDMY	SPLW1	SPLW0	—	—	—	—	—
Initial Value	0	0	1	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	TXDMY	0	R/W	Dummy Data Transmission Enable Enables or disables dummy data transmission. When communication is performed with this bit set to 1, dummy data is transmitted from the MOSI pin and a serial communication can be performed even if there is no transmit data in the transmit buffer. Specifically, if there is no transmit data in the transmit buffer and this bit is set to 1, dummy data is transferred to the shift register. Data previously transmitted from the pin is used as dummy data. If this bit is set to 1 after the initialization and a transfer is performed, the transmitted dummy data is undefined. 0: Disables dummy data transmission. 1: Enables dummy data transmission. *Note*: This bit is valid only in the master mode.
6	SPLW1	0	R/W	Access Width Specification Specifies the width for accessing the data register (SPDR). If the length of data transferred to SPDR does not agree with these bit settings, operation is not guaranteed.*[1] 00: Setting prohibited 01: SPDR is accessed in bytes (8 bits). 10: SPDR is accessed in words (16 bits). 11: SPDR is accessed in longwords (32 bits).
5	SPLW0	1	R/W	
4 to 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Note 1. The data length is specified by the SPB3 to SPB0 bits in the command register (SPCMD). See **Section 24.3.5, Data Register (SPDR)**.

24.3.10 Clock Delay Register (SPCKD)

SPCKD sets a period from the beginning of SSL signal assertion to RSPCK oscillation (RSPCK delay) when the SCKDEN bit in the command register (SPCMD) is 1. If the contents of SPCKD are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set 000b to SCKDL2 to SCKDL0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKDL 2	SCKDL 1	SCKDL 0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SCKDL2	0	R/W	RSPCK Delay Setting These bits set an RSPCK delay value when the SCKDEN bit in SPCMD is 1. The relationship between the setting of SCKDL2 to SCKDL0 and the RSPCK delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK
1	SCKDL1	0	R/W	
0	SCKDL0	0	R/W	

24.3.11 Slave Select Negation Delay Register (SSLND)

SSLND sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSL signal during a serial transfer by this module in master mode. If the contents of SSLND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set 000b to SLNDL2 to SLNDL0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SLNDL2	0	R/W	SSL Negation Delay Setting These bits set an SSL negation delay when the SLNDEN bit in SPCMD is 1. The relationship between the setting of SLNDL2 to SLNDL0 and the SSL negation delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK
1	SLNDL1	0	R/W	
0	SLNDL0	0	R/W	

24.3.12 Next-Access Delay Register (SPND)

SPND sets a non-active period (next-access delay) after termination of a serial transfer when the SPNDEN bit in the command register (SPCMD) is 1. If the contents of SPND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set 000b to SPNDL2 to SPNDL0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPNDL 2	SPNDL 1	SPNDL 0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SPNDL2	0	R/W	Next-Access Delay Setting These bits set a next-access delay when the SPNDEN bit in SPCMD is 1. The relationship between the setting of SPNDL2 to SPNDL0 and the next-access delay value is shown below. 000: 1 RSPCK + 2 P0ϕ 001: 2 RSPCK + 2 P0ϕ 010: 3 RSPCK + 2 P0ϕ 011: 4 RSPCK + 2 P0ϕ 100: 5 RSPCK + 2 P0ϕ 101: 6 RSPCK + 2 P0ϕ 110: 7 RSPCK + 2 P0ϕ 111: 8 RSPCK + 2 P0ϕ
1	SPNDL1	0	R/W	
0	SPNDL0	0	R/W	

24.3.13 Command Register (SPCMD)

Each channel has four command registers (SPCMD0 to SPCMD3). SPCMD0 to SPCMD3 are used to set a transfer format for master mode operation. Some of the bits in SPCMD0 are used to set a transfer mode for slave mode operation. In master mode, this module sequentially references SPCMD0 to SPCMD3 according to the settings in bits SPSLN1 and SPSLN0 in the sequence control register (SPSCR), and executes the serial transfer that is set in the referenced SPCMD. While bit TEND in the status register (SPSR) indicates that transmission is not completed, correct operation of this module cannot be guaranteed if SPCMD is changed that is referred by this module. SPCMD referenced by this module in master mode can be checked by means of bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the function of this module in slave mode is enabled, operation cannot be guaranteed if the value set in SPCMD0 is changed.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
Initial Value	0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	SCKDEN	0	R/W	RSPCK Delay Setting Enable Sets the period from the point this module in master mode activates the SSL signal until the RSPCK starts oscillation (RSPCK delay). If the SCKDEN bit is 0, this module sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, this module starts the oscillation of RSPCK at an RSPCK delay in compliance with the clock delay register (SPCKD) settings. To use this module in slave mode, the SCKDEN bit should be set to 0. 0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay equal to SPCKD settings.
14	SLNDEN	0	R/W	SSL Negation Delay Setting Enable Sets the period from the point this module in master mode stops RSPCK oscillation until this module sets the SSL signal inactive (SSL negation delay). If the SLNDEN bit is 0, this module sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is 1, this module negates the SSL signal at an SSL negation delay in compliance with the slave select negation delay register (SSLND) settings. To use this module in slave mode, the SLNDEN bit should be set to 0. 0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay equal to SSLND settings.
13	SPNDEN	0	R/W	Next-Access Delay Enable Sets the period from the point this module in master mode terminates a serial transfer and sets the SSL signal inactive until this module enables the SSL signal assertion for the next access (next-access delay). If the SPNDEN bit is 0, this module sets the next-access delay to 1 RSPCK + 2P0ϕ. If the SPNDEN bit is 1, this module inserts a next-access delay in compliance with the next-access delay register (SPND) settings. To use this module in slave mode, the SPNDEN bit should be set to 0. 0: A next-access delay of 1 RSPCK + 2 P0ϕ 1: A next-access delay equal to SPND settings.
12	LSBF	0	R/W	LSB First Sets the data format in master mode or slave mode to MSB first or LSB first. 0: MSB first 1: LSB first

Bit	Bit Name	Initial Value	R/W	Description
11	SPB3	0	R/W	Data Length Setting These bits set a transfer data length in master mode or slave mode. 0100 to 0111: 8 bits 1111: 16 bits 0010, 0011: 32 bits Others: Setting prohibited
10	SPB2	1	R/W	
9	SPB1	1	R/W	
8	SPB0	1	R/W	
7	SSLKP	0	R/W	SSL Signal Level Keeping When this module in master mode performs a serial transfer, this bit specifies whether the SSL signal level for the current command is to be kept or negated between the SSL negation timing associated with the current command and the SSL assertion timing associated with the next command. To use this module in slave mode, the SSLKP bit should be set to 0. 0: Negates the SSL signal upon completion of transfer. 1: Keeps the SSL signal level from the end of the transfer until the beginning of the next access.
6 to 4	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
3	BRDV1	1	R/W	Bit Rate Division Setting These bits are used to determine the bit rate. A bit rate is determined by combinations of bits BRDV1 and BRDV 0 and the settings in the bit rate register (SPBR) (see **Section 24.3.8, Bit Rate Register (SPBR)**). The settings in SPBR determine the base bit rate. The settings in bits BRDV1 and BRDV0 are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In the bits SPCMD0 to SPCMD3, different BRDV1 and BRDV0 settings can be specified. This permits the execution of serial transfers at a different bit rate for each command. 00: Select the base bit rate. 01: Select the base bit rate divided by 2. 10: Select the base bit rate divided by 4. 11: Select the base bit rate divided by 8.
2	BRDV0	1	R/W	
1	CPOL	0	R/W	RSPCK Polarity Setting Sets an RSPCK polarity in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK polarity should be set for both modules. 0: RSPCK = 0 when idle 1: RSPCK = 1 when idle
0	CPHA	1	R/W	RSPCK Phase Setting Sets an RSPCK phase in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK phase should be set for both modules. 0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge

24.3.14 Buffer Control Register (SPBFCR)

SPBFCR resets the number of data units in the transmit buffer (SPTX) or receive buffer (SPRX) and sets the number of triggering data units.

Bit	7	6	5	4	3	2	1	0
	TXRST	RXRST	TXTRG[1:0]		—	RXTRG[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TXRST	0	R/W	Transmit Buffer Data Reset Resets the transmit buffer to an empty state. Transmit data in the transmit buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation.*[1] 1: Enables the reset operation *Note 1.* The reset operation is performed after a power-on reset.
6	RXRST	0	R/W	Receive Buffer Data Reset Resets the receive buffer to an empty state. Receive data in the receive buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation.*[1] 1: Enables the reset operation *Note 1.* The reset operation is performed after a power-on reset.
5, 4	TXTRG[1:0]	00	R/W	Transmit Buffer Data Triggering Number*[2] Specifies the timing at which the transmit buffer empty state is determined, that is when the SPTEF flag in the status register is set. When the number of bytes of data in the transmit buffer (SPTX) is equal to or less than the specified triggering number, the SPTEF flag is set to 1. 00: 7 bytes (1)*[1] 01: 6 bytes (2)*[1] 10: 4 bytes (4)*[1] 11: 0 bytes (8)*[1] *Note 1.* The value in the parenthesis shows the number of available bytes in the transmit buffer (SPTX). *Note 2.* When transferring the data by using DMA, set the following values according to the data transfer length. (The data transfer length is set in SPBn (n = 0-3) of the command register (SPCMD)). 1) Data transfer length is 1 byte SPBFCR (TXTRG): 01 (6 bytes) or 10 (4 byte) or 11 (0 byte) 2) Data transfer length is 2 bytes SPBFCR (TXTRG): 10 (4 bytes) or 11 (0 byte) 3) Data transfer length is 4bytes SPBFCR (TXTRG): 11 (0 byte)
3	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	RXTRG[2:0]	000	R/W	Receive Buffer Data Triggering Number Specifies the timing at which the receive buffer full state is determined, that is when the SPRF flag in the status register is set. When the number of bytes of data in the receive buffer (SPRX) is equal to or greater than the specified triggering number, the SPRF flag is set to 1. 000: 1 byte (31)*1 001: 2 bytes (30)*1 010: 4 bytes (28)*1 011: 8 bytes (24)*1 100: 16 bytes (16)*1 101: 24 bytes (8)*1 110: 32 bytes (0)*1 111: 5 bytes (27)*1 *Note 1.* The value in the parenthesis shows the number of available bytes in the receive buffer (SPRX).

24.3.15 Buffer Data Count Setting Register (SPBFDR)

SPBFDR indicates the number of data units stored in the transmit buffer (SPTX) and receive buffer (SPRX). The upper eight bits indicate the number of transmit data units in SPTX and the lower eight bits indicate the number of receive data units in SPRX.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	T[3:0]				—	—	R[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
11 to 8	T[3:0]	0000	R	Indicates the number of bytes of data to be transmitted in SPTX. 0000b indicates that SPTX is empty. 1000b indicates that SPTX is full.
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5 to 0	R[5:0]	000000	R	Shows the number of bytes of received data in SPRX. 000000b indicates that SPRX is empty. 100000b indicates that SPRX is full.

24.4 Operation

In this section, the serial transfer period means a period from the beginning of driving valid data to the fetching of the final valid data.

24.4.1 Overview of Operations

This module is capable of serial transfers in slave mode and master mode. A particular mode of this module can be selected by using the MSTR bit in the control register (SPCR). **Table 24.4** gives the relationship between the modes and SPCR settings, and a description of each mode.

Table 24.4 Relationship between Modes and SPCR and Description of Each Mode

Mode	Slave (SPI Operation)	Master (SPI Operation)
MSTR bit setting	0	1
MODFEN bit setting	0 or 1	0
RSPCK signal	Input	Output
MOSI signal	Input	Output
MISO signal	Output/Hi-Z	Input
SSL signal	Input	Output
SSL polarity modification function	Supported	Supported
Transfer rate	Up to P0ϕ/8	Up to P0ϕ/2 (Up to P0ϕ/4 when P0ϕ is 100 MHz)
Clock source	RSPCK input	On-chip baud rate generator
Clock polarity	Two	Two
Clock phase	Two	Two
First transfer bit	MSB/LSB	MSB/LSB
Transfer data length	8, 16, or 32 bits	8, 16, or 32 bits
Burst transfer	Possible (CPHA = 1)	Possible (CPHA = 0, 1)
RSPCK delay control	Not supported	Supported
SSL negation delay control	Not supported	Supported
Next-access delay control	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmit buffer is written when SPE = 1
Sequence control	Not supported	Supported
Transmit buffer empty detection	Supported	Supported
Receive buffer full detection	Supported	Supported
Overrun error detection	Supported	Not Supported
Mode fault error detection	Supported (MODFEN = 1)	Not supported

24.4.2 Pin Control

According to the MSTR bit in the control register (SPCR), this module can automatically switch pin directions and output modes. **Table 24.5** shows the relationship between pin states and bit settings.

Table 24.5 Relationship between Pin States and Bit Settings

Mode	Pin	Pin State*[1]
Master mode (SPI operation) (MSTR = 1)	RSPCK	CMOS output
	SSL	CMOS output
	MOSI	CMOS output
	MISO	Input
Slave mode (SPI operation) (MSTR = 0)	RSPCK	Input
	SSL	Input
	MOSI	Input
	MISO*[1]	CMOS output/Hi-Z

Note 1. When SSL is at the non-active level or the SPE bit in SPCR is cleared to 0, the pin state is Hi-Z.

This module in master mode (SPI operation) determines MOSI signal values during the SSL negation period (including the SSL retention period during a burst transfer) according to MOIFE and MOIFV bit settings in SPPCR, as shown in **Table 24.6**.

Table 24.6 MOSI Signal Value Determination during SSL Negation Period

MOIFE	MOIFV	MOSI Signal Value during SSL Negation Period
0	0, 1	Last output value from previous transfer (The value is undefined when CPHA is 0)
1	0	Always 0
1	1	Always 1

24.4.3 System Configuration Example

(1) Master/Slave (with This LSI Acting as Master)

Figure 24.2 shows a master/slave system configuration example when this LSI is used as a master. In master/slave configuration, the SSL output of this LSI (master) is not used. The SSL input of the slave is fixed to the low level, and the slave is always maintained in a selected state. In the transfer format corresponding to the case where the CPHA bit in the control register (SPCR) is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSL output of this LSI should be connected to the SSL input of the slave device.

This LSI (master) always drives the RSPCK and MOSI. The slave always drives the MISO.

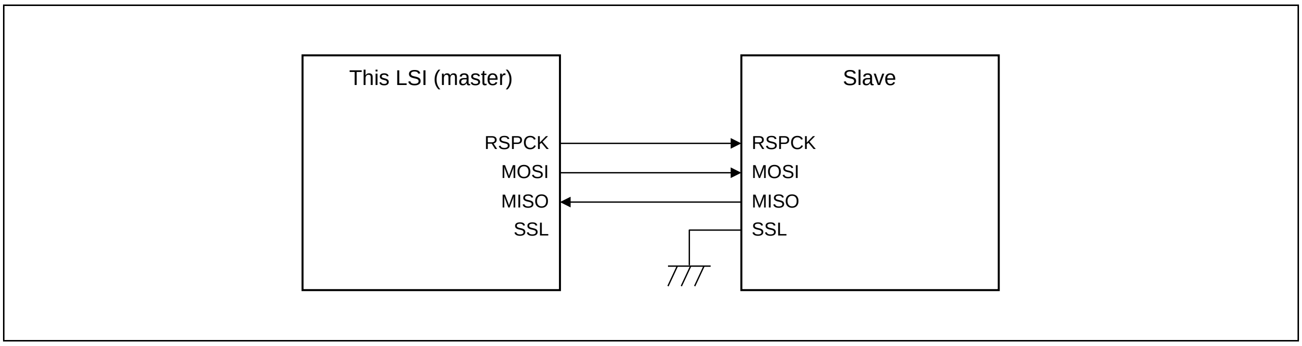


Figure 24.2 Master/Slave Configuration Example (This LSI = Master)

(2) Master/Slave (with This LSI Acting as Slave)

Figure 24.3 shows a master/slave system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSL pin is used as SSL input. The master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISO. When SSL is at the non-active level, the pin state is Hi-Z.

In the slave configuration in which the CPHA bit in the command register (SPCMD) is set to 1, the SSL input of this LSI (slave) is fixed to the 0 level, this LSI (slave) is always maintained in a selected state, and in this manner it is possible to execute serial transfer (**Figure 24.4**).

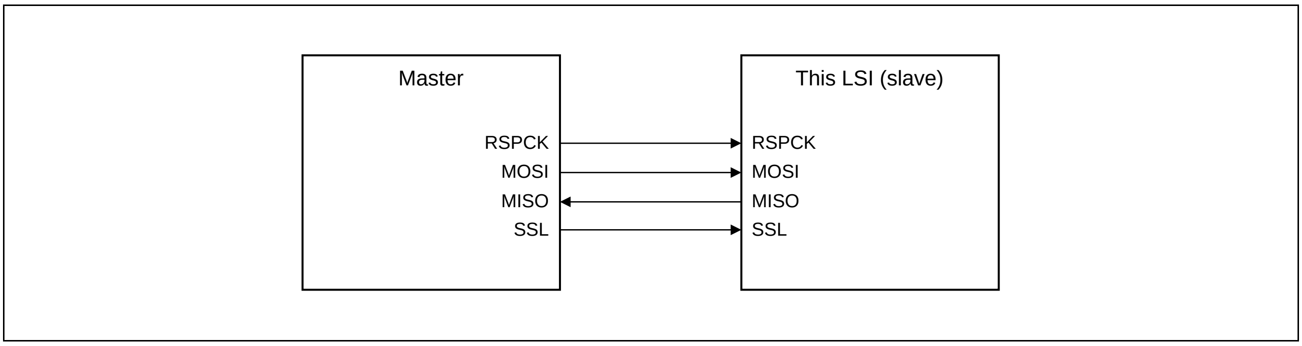


Figure 24.3 Master/Slave Configuration Example (This LSI = Slave)

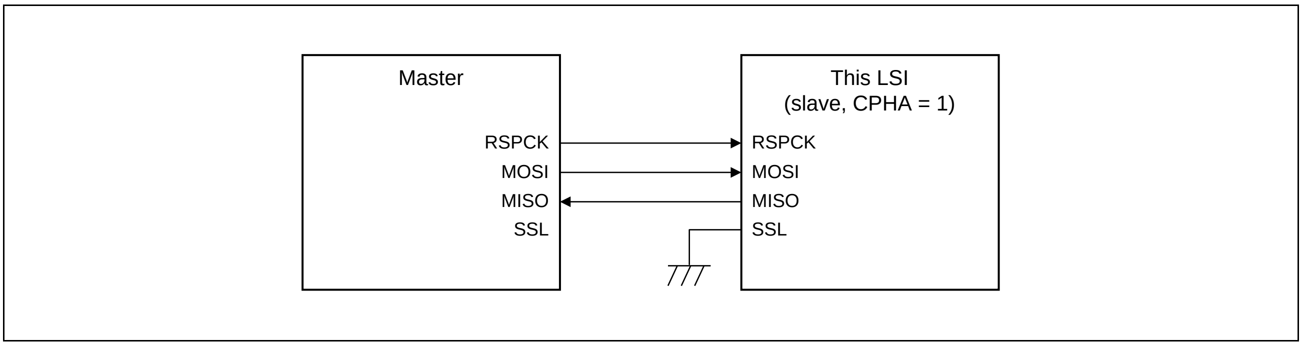


Figure 24.4 Master/Slave Configuration Example (This LSI = Slave, CPHA = 1)

(3) Master/Multi-Slave (with This LSI Acting as Slave)

Figure 24.5 shows a master/multi-slave system configuration example when this LSI is used as a slave. In the example of **Figure 24.5**, the system is comprised of a master and two LSIs (slave X and slave Y).

The RSPCK and MOSI outputs of the master are connected to the RSPCK and MOSI inputs of the LSIs (slave X and slave Y). The MISO outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the master. SSLX and SSLY outputs of the master are connected to the SSL inputs of the LSIs (slave X and slave Y), respectively.

The master always drives RSPCK, MOSI, SSLX, and SSLY. Of the LSIs (slave X and slave Y), the slave that receives low level input into the SSL0 input drives MISO.

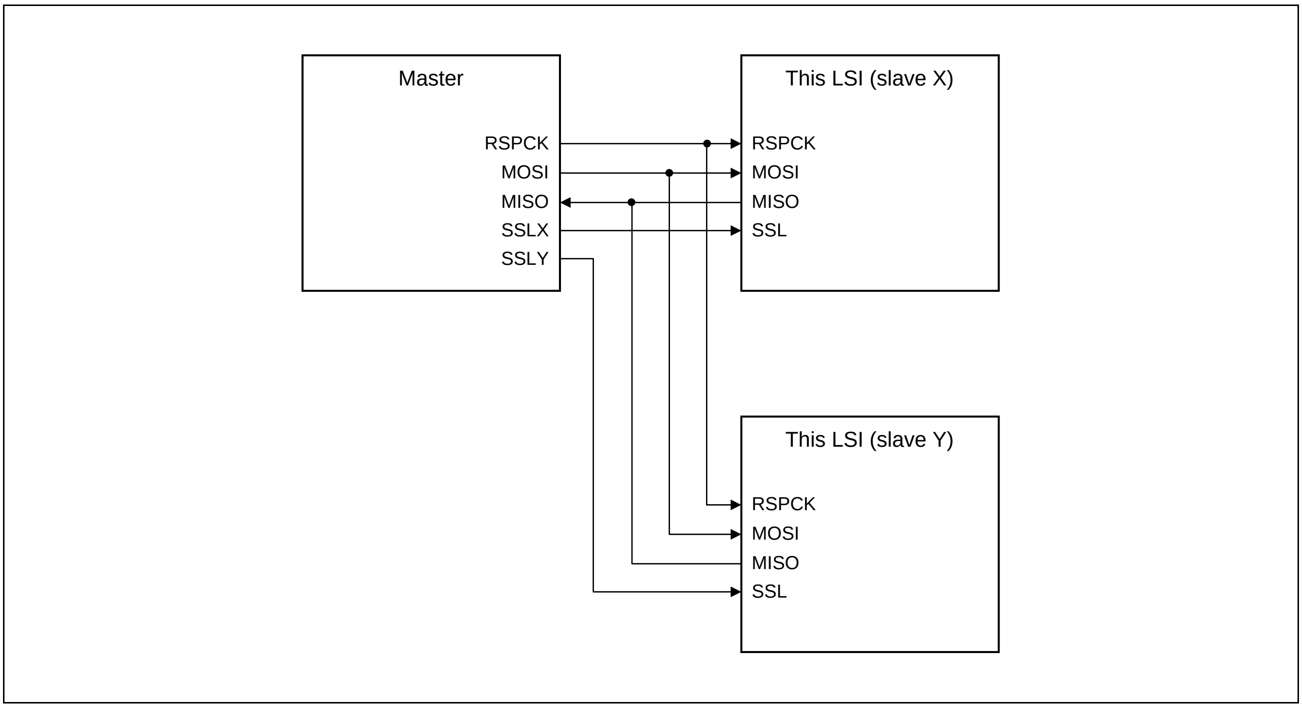


Figure 24.5 Master/Multi-Slave Configuration Example (This LSI = Slave)

24.4.4 Transfer Format

(1) CPHA = 0

Figure 24.6 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 0. In **Figure 24.6**, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the settings of this module. For details, see **Section 24.4.2, Pin Control**.

When the CPHA bit is 0, the driving of valid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The first RSPCK signal change timing that occurs after the SSL signal assertion becomes the first transfer data fetching timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSI and MISO signals is always 1/2 RSPCK cycle after the transfer data fetch timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1 denotes a period from an SSL signal assertion to RSPCK oscillation (RSPCK delay). t2 denotes a period from the cessation of RSPCK oscillation to an SSL signal negation (SSL negation delay). t3 denotes a period in which SSL signal assertion is suppressed for the next transfer after the end of serial transfer (next-access delay). t1, t2, and t3 are controlled by a master device running on the system. For a description of t1, t2, and t3 when this module is in master mode, see **Section 24.4.3(1), Master/Slave (with This LSI Acting as Master)**.

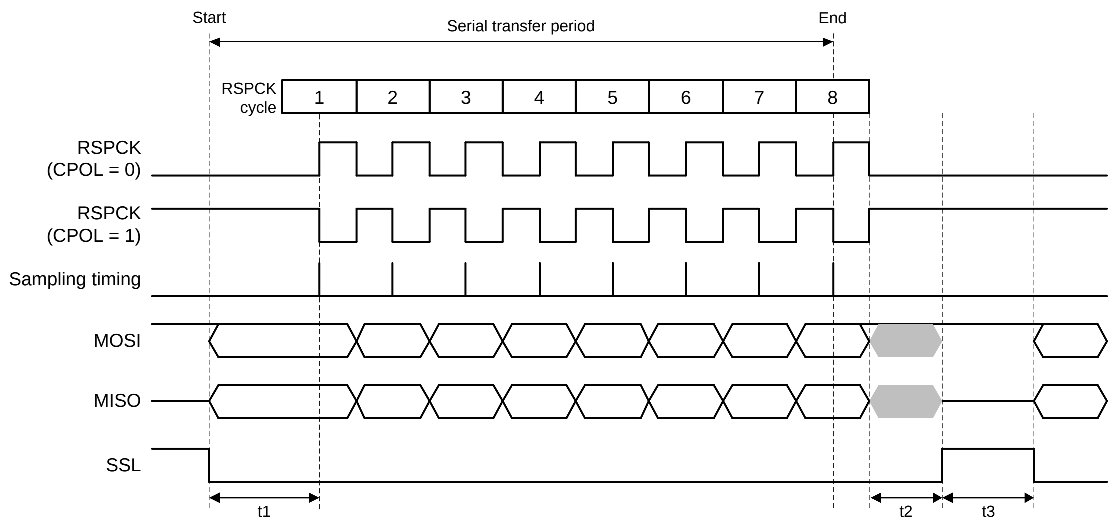


Figure 24.6 Transfer Format (CPHA = 0)

(2) CPHA = 1

Figure 24.7 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 1. In **Figure 24.7**, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the modes (master or slave). For details, see **Section 24.4.2, Pin Control**.

When the CPHA bit is 1, the driving of invalid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The driving of valid data to the MOSI and MISO signals commences at the first RSPCK signal change timing that occurs after the SSL signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycle after the data update timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1, t2, and t3 are the same as those in the case of CPHA = 0. For a description of t1, t2, and t3 when this module is in master mode, see **Section 24.4.3(1), Master/Slave (with This LSI Acting as Master)**.

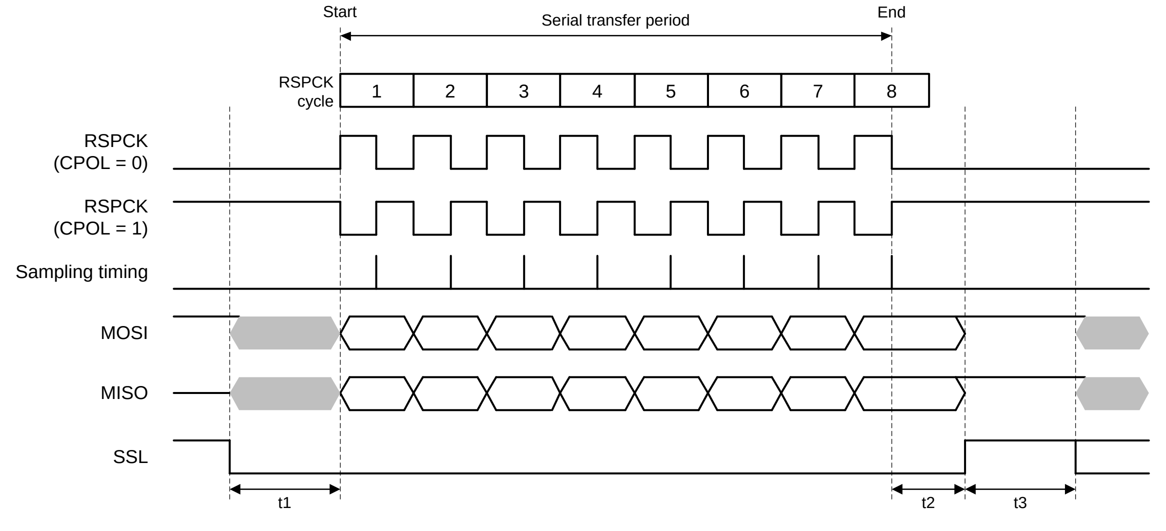


Figure 24.7 Transfer Format (CPHA = 1)

24.4.5 Data Format

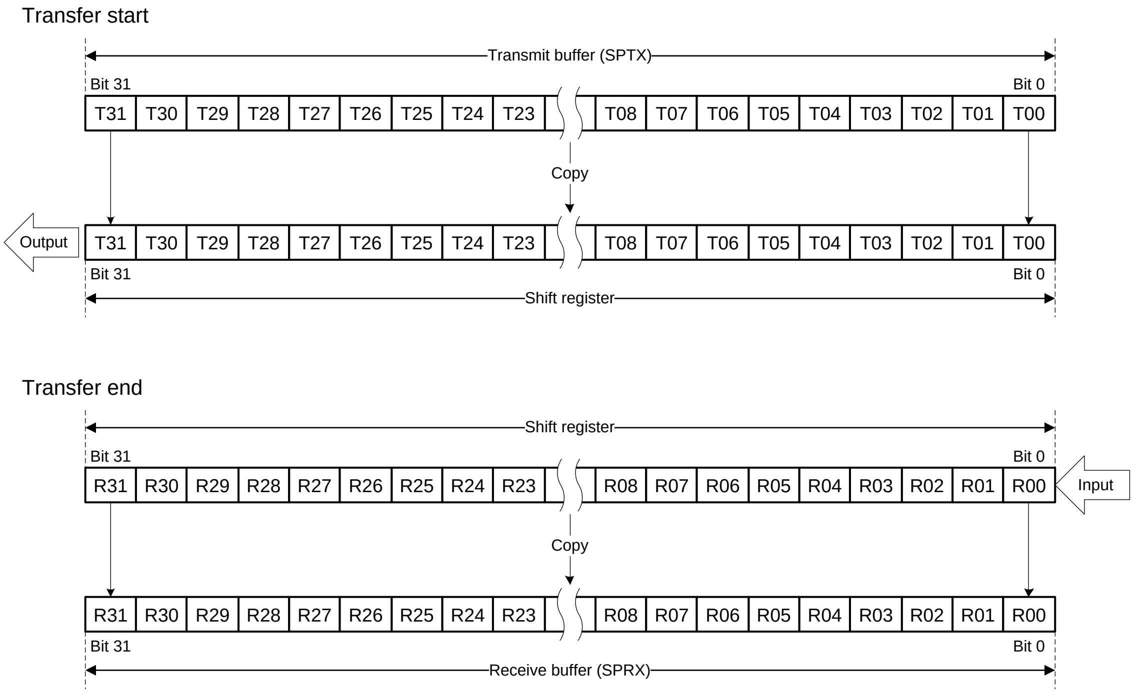
The data format depends on the settings in the command register (SPCMD). Irrespective of MSB/LSB first, this module treats the range from the LSB of the data register (SPDR) to the assigned data length as transfer data.

(1) MSB First Transfer (32-Bit Data)

Figure 24.8 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer of SPDR. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to R00 is stored in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R31 to R00 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

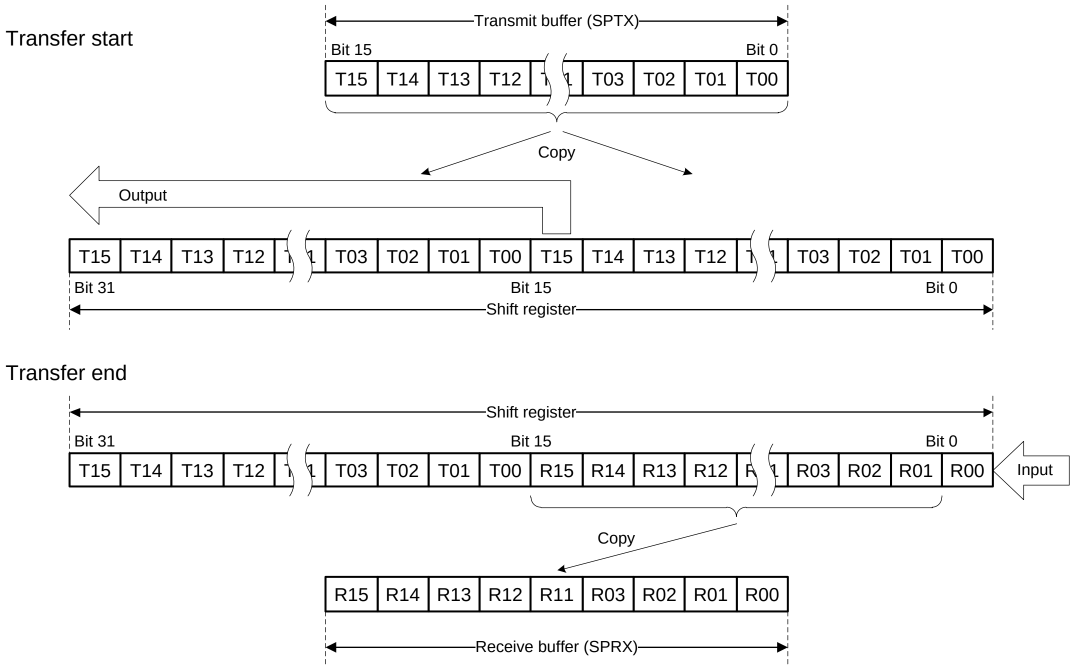
Figure 24.8 MSB First Transfer (32-Bit Data)

(2) MSB First Transfer (16-Bit Data)

Figure 24.9 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 15 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R15 to R00 is stored in bits 15 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 16 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R15 to R00 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

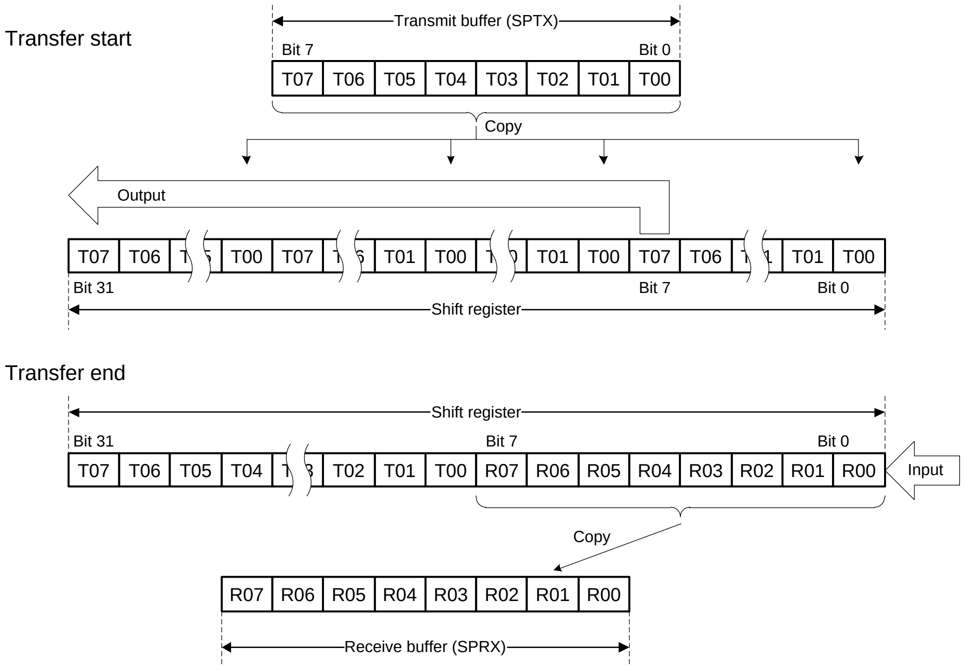
Figure 24.9 MSB First Transfer (16-Bit Data)

(3) MSB First Transfer (8-Bit Data)

Figure 24.10 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 7 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R07 to R00 is stored in bits 7 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 8 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary area in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R07 to R00 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MOSI (master)/MISO (slave)

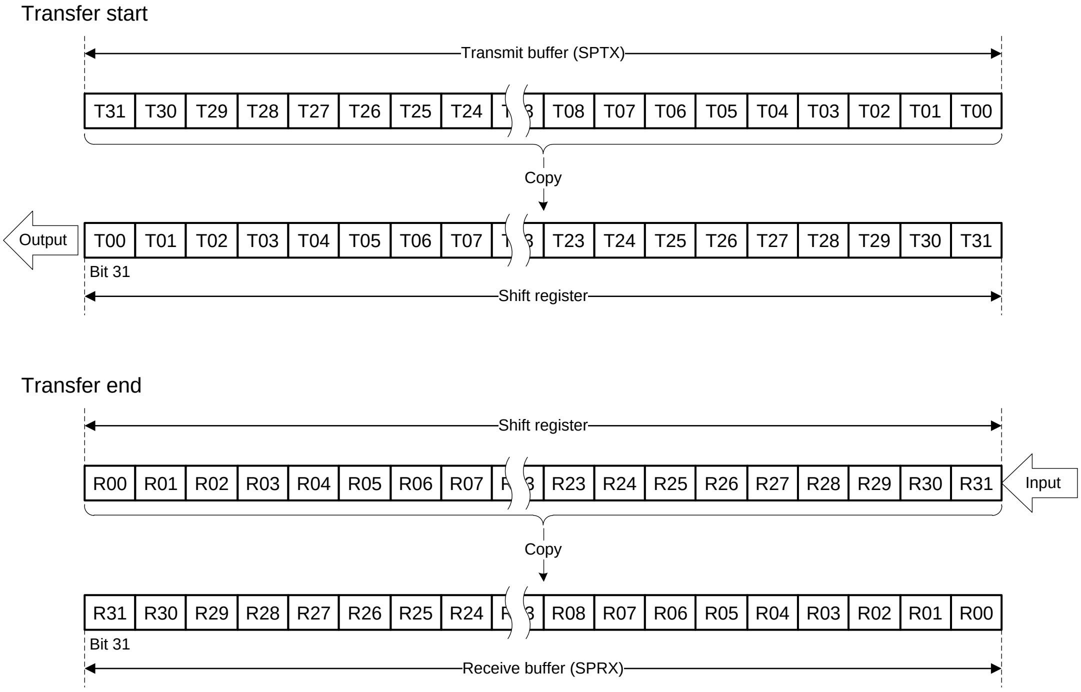
Figure 24.10 MSB First Transfer (8-Bit Data)

(4) LSB First Transfer (32-Bit Data)

Figure 24.11 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to R31 is stored in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of the SPDR, received data R00 to R31 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

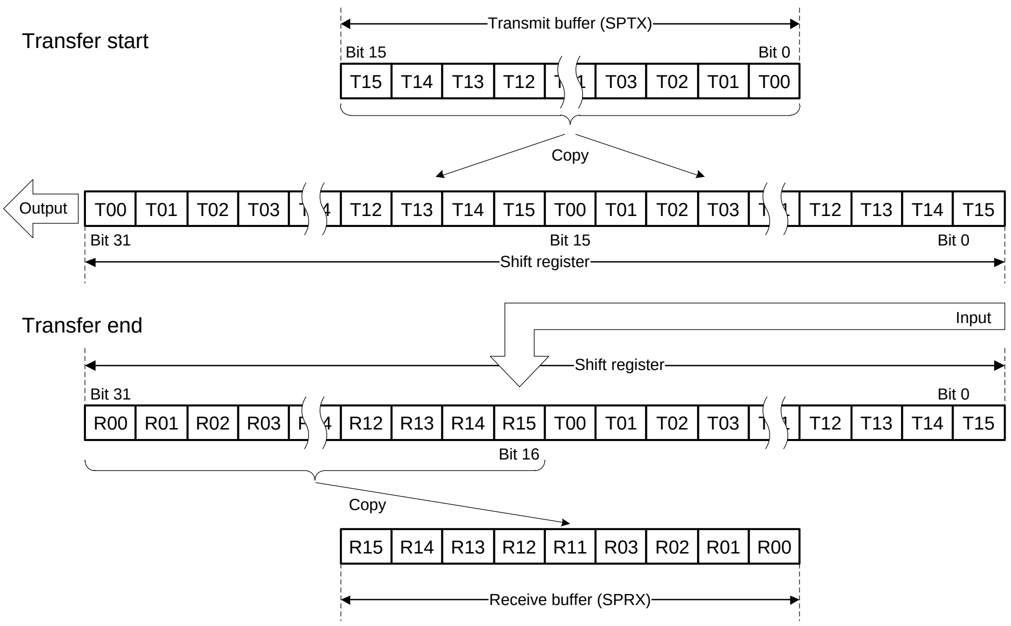
Figure 24.11 LSB First Transfer (32-Bit Data)

(5) LSB First Transfer (16-Bit Data)

Figure 24.12 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 16 in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R00 to R15 is stored in bits 31 to 16 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 15 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R15 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

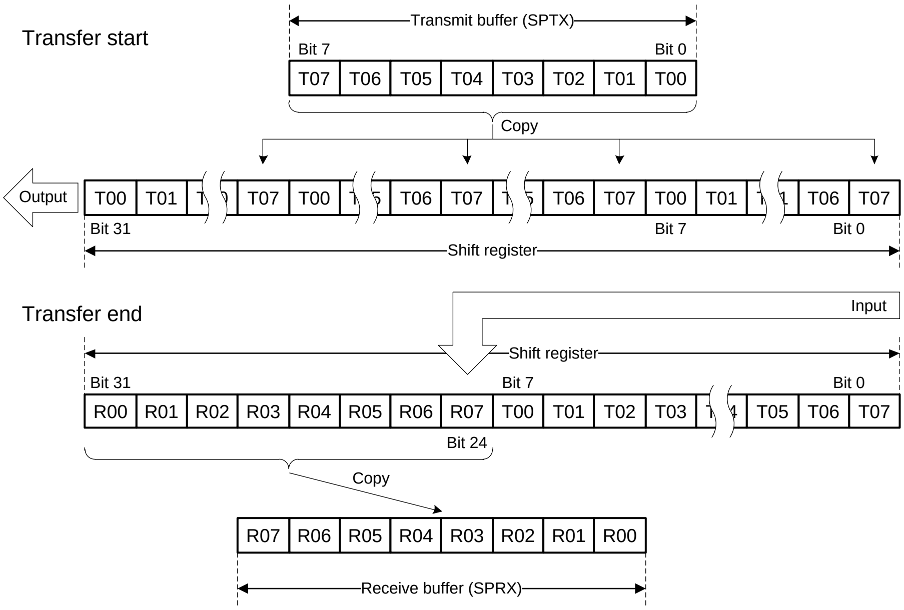
Figure 24.12 LSB First Transfer (16-Bit Data)

(6) LSB First Transfer (8-Bit Data)

Figure 24.13 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 24 in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R00 to R07 is stored in bits 31 to 24 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 23 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R07 is shifted out from the shift register.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 24.13 LSB First Transfer (8-Bit Data)

24.4.6 Error Detection

In the normal serial transfer, the data written to the transmit buffer of the data register (SPDR) is serially transmitted, and the serially received data can be read from the receive buffer of SPDR. If access is made to SPDR, depending on the status of the transmit buffer/receive buffer or the status at the beginning or end of serial transfer, in some cases non-normal transfers can be executed.

If a non-normal transfer operation occurs, this module detects the event as an overrun error or a mode fault error. **Table 24.7** shows the relationship between non-normal transfer operations and the error detection function.

Table 24.7 Relationship between Non-Normal Transfer Operations and Error Detection Function

	Occurrence Condition	Operation	Error Detection
A	SPDR is written when the transmit buffer is full.	Missing write data.	None
B	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is serially transmitted.	None
C	SPDR is read when the receive buffer is empty.	The output data is undefined.	None
D	Serial transfer terminates when the receive buffer is full.	Missing serial receive data.	Overrun error (only in slave mode)
E	The SSL input signal is negated during serial transfer in slave mode.	Serial transfer suspended. Missing send/receive data. Operation disabled.	Mode fault error

On operation A shown in **Table 24.7**, this module does not detect an error. Whether SPDR can be written to or not can be checked using the T[3:0] bits in the buffer data count setting register (SPBFDR).

Likewise, this module does not detect an error on operation B. In a serial transfer that was started before the shift register was updated, this module sends the data that was received in the previous serial transfer, and does not treat the operation indicated in B as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of SPDR, thus it can be correctly read.

Similarly, this module does not detect an error on operation C. To prevent extraneous data from being read, the number of receive data units stored in the receive buffer should be read from the R[5:0] bits in the buffer data count setting register (SPBFDR).

An overrun error shown in D is described in **Section 24.4.6(1), Overrun Error**. A mode fault error shown in E is described in **Section 24.4.6(2), Mode Fault Error**.

(1) Overrun Error

If serial transfer ends when the receive buffer of the data register (SPDR) is full, this module detects an overrun error, and sets the OVRF bit in SPSR to 1. When the OVRF bit is 1, this module does not copy data from the shift register to the receive buffer so that the data prior to the occurrence of the error is retained in the receive buffer. To reset the OVRF bit in SPSR to 0, either perform a power-on reset, or write a 0 to the OVRF bit after SPSR has been read with the OVRF bit set to 1.

Figure 24.14 shows an example of operation of the SPRF and OVRF bits in SPSR. The SPSR and SPDR accesses shown in **Figure 24.14** indicate the condition of accesses to SPSR and SPDR, respectively, where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of **Figure 24.14**, this module performs an 8-bit serial transfer in which the CPHA bit in the command register (SPCMD) is 1, and CPOL is 0. The numbers given under the RSPCK waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

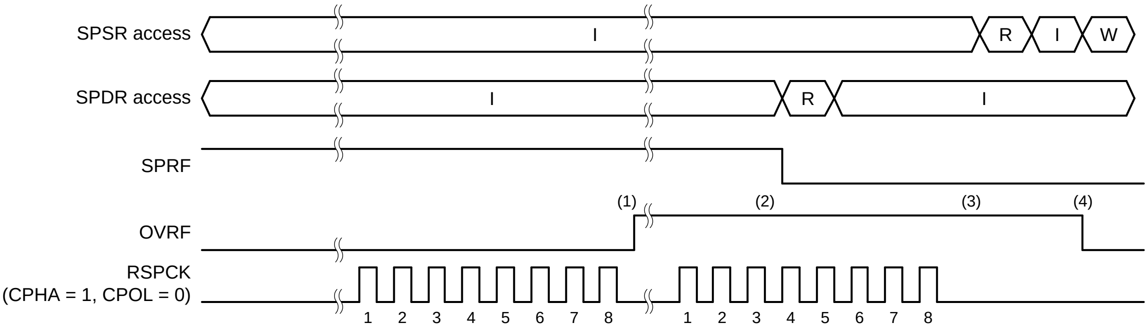


Figure 24.14 SPRF and OVRF Bit Operation Example

The operation of the flags at the timing shown in steps (1) to (4) in the figure is described below.

1. If a serial transfer terminates when the receive buffer does not have a space for the receive data length, this module detects an overrun error, and sets the OVRF bit to 1. This module does not copy the data in the shift register to the receive buffer.
2. The OVFR bit is not cleared even when SPDR is read and thus the number of data bytes in the receive buffer becomes less than the number of the receive buffer data triggering number specified by the RXTRG bits.
3. If the serial transfer terminates in an overrun error state, this module determines that the shift register is empty; in this manner, data transfer is enabled from the transmit buffer to the shift register.
4. If 0 is written to the OVRF bit after SPSR is read with OVRF = 1, this module clears the OVRF bit.

The occurrence of an overrun can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. When executing a serial transfer without using an error interrupt, measures should be taken to ensure the early detection of overrun errors, such as reading SPSR immediately after SPDR is read.

The OVRF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition in which the OVRF bit is set to 1, 0 is written to the OVRF bit.
- Power-on reset

NOTE

When the receive buffer has area enough to store receive data with an overrun error, this module receives receive data.

(2) Mode Fault Error

When the MSTR bit is 0, this module operates in slave mode. This module detects a mode fault error if the SSL input signal is negated during the serial transfer period (from the time the driving of valid data is started to the time the final valid data is fetched) when the MODFEN bit is 1 in slave mode.

Upon detecting a mode fault error, this module stops driving of the output signals and clears the SPE bit in SPCR to 0. When the SPE bit is cleared to 0, the function of this module is disabled, and this module stops driving external signals. For details of disabling the function of this module by clearing the SPE bit to 0, see **Section 24.4.7, Initialization**.

The occurrence of a mode fault error can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. To detect a mode fault error without using an error interrupt, it is necessary to poll SPSR.

When the MODF bit is 1, writing 1 to the SPE bit is ignored. To enable the function of this module after the detection of a mode fault error, the MODF bit must be set to 0. The MODF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition where the MODF bit has turned 1, 0 is written to the MODF bit.
- Power-on reset

24.4.7 Initialization

If 0 is written to the SPE bit in the control register (SPCR) or this module clears the SPE bit to 0 because of the detection of a mode fault error, this module disables the module function, and initializes a part of the module function. When a power-on reset is generated, this module initializes all of the module function. An explanation follows of initialization by the clearing of the SPE bit.

(1) Initialization by Clearing SPE Bit

When the SPE bit in SPCR is cleared, this module performs the following initialization:

- Suspending any serial transfer that is being executed
- Stopping the driving of output signals (Hi-Z) in slave mode
- Initializing the internal state
- Initializing the TEND bit in SPSR

Initialization by the clearing of the SPE bit does not initialize the control bits of this module. For this reason, this module can be started in the same transfer mode as prior to the initialization if the SPE bit is re-set to 1.

24.4.8 SPI Operation

(1) Multi-Master Mode Operation

This section explains the operation in multi-master mode.

(a) Starting Serial Transfer

A serial transfer is started when transmit data is copied from the transmit buffer to the shift register, the shift register becomes full, and the receive buffer has a space for the receive data length. If transmit data has already been written to the shift register, data is not copied from the transmit buffer to the shift register.

For details of the transfer format, see **Section 24.4.4, Transfer Format**.

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register (SPCMD), this module terminates the serial transfer after transmitting an RSPCK edge corresponding to the final sampling timing. After the serial transfer is completed, receive data is copied from the shift register to the receive buffer. If the receive buffer does not have a space for the receive data length after receive data is copied from the shift register to the receive buffer, another serial transfer will not be performed. In order to perform another serial transfer, data for the receive data length should be read from the receive buffer to secure the space for the receive data.

It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the data length depends on the settings in bits SPB3 to SPB0 in SPCMD. For details on the transfer format, see **Section 24.4.4, Transfer Format**.

(c) Sequence Control

The transfer format that is employed in master mode is determined by the sequence control register (SPSCR), command registers 0 to 3 (SPCMD0 to SPCMD3), the bit rate register (SPBR), the clock delay register (SPCKD), the slave select negation delay register (SSLND), and the next-access delay register (SPND).

SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by this module in master mode. The following items are set in command registers SPCMD0 to SPCMD3: SSL output signal value, MSB/ LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether SPCKD is to be referenced, whether SSLND is to be referenced, and whether SPND is to be referenced. SPBR holds some of the bit rate settings; SPCKD, a clock delay value; SSLND, an SSL negation delay; and SPND, a next-access delay value.

According to the sequence length that is assigned to SPSCR, this module makes up a sequence comprised of a part or all of SPCMD0 to SPCMD3. This module contains a pointer to the SPCMD that makes up the sequence. The value of this pointer can be checked by reading bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the SPE bit in the control register (SPCR) is set to 1 and the function of this module is enabled, this module loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. This module increments the pointer each time the next-access delay period for a data transfer ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, this module sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.

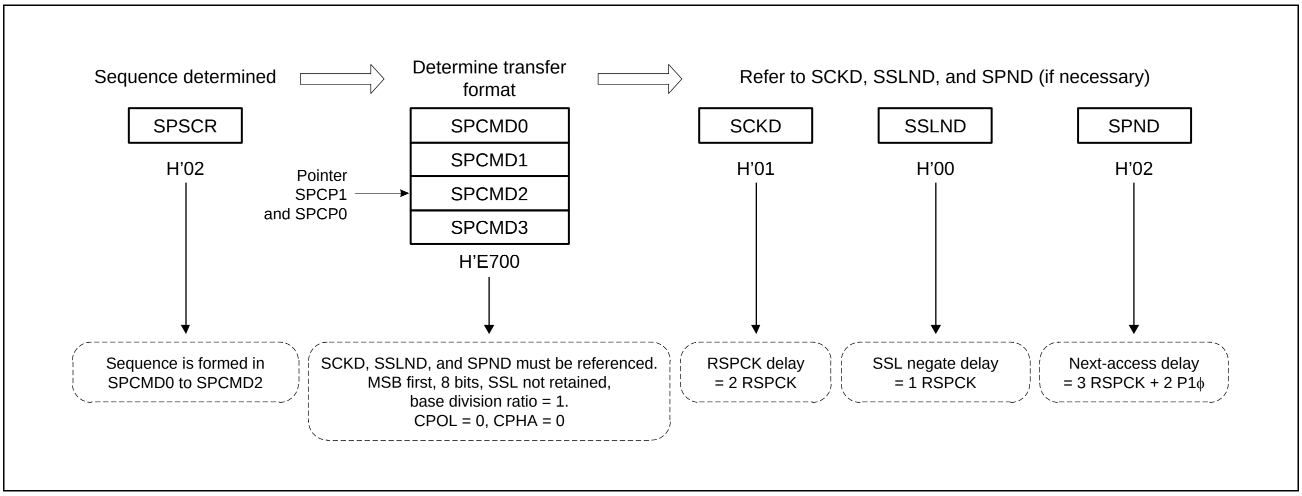


Figure 24.15 Determination Procedure of Serial Transfer Mode in Master Mode

(d) Burst Transfer

If the SSLKP bit in the command register (SPCMD) that this module references during the current serial transfer is 1, this module keeps the SSL signal level during the serial transfer until the beginning of the SSL signal assertion for the next serial transfer. If the SSL signal level for the next serial transfer is the same as the SSL signal level for the current serial transfer, this module can execute continuous serial transfers while keeping the SSL signal assertion status (burst transfer).

Figure 24.16 shows an example of an SSL signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 settings. The text below explains operations (1) to (7) as depicted in **Figure 24.16**. It should be noted that the polarity of the SSL output signal depends on the settings in the slave select polarity register (SSLP).

1. Based on SPCMD0, this module asserts the SSL signal and inserts RSPCK delays.
2. Serial transfers are executed according to SPCMD0.
3. SSL negation delays are inserted.
4. Because the SSLKP bit in SPCMD0 is 1, this module keeps the SSL signal value on SPCMD0. This period is sustained, at the shortest, for a period equal to the next-access delay of SPCMD0. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
5. Based on SPCMD1, this module asserts the SSL signal and inserts RSPCK delays.
6. Serial transfers are executed according to SPCMD1.
7. Because the SSLKP bit in SPCMD1 is 0, this module negates the SSL signal. In addition, a next-access delay is inserted according to SPCMD1.

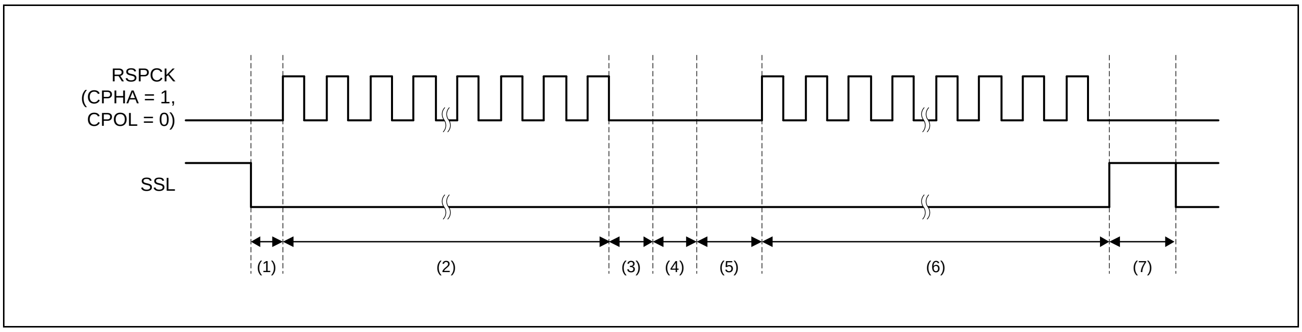


Figure 24.16 Example of Burst Transfer Operation using SSLKP Bit

If the SSL signal settings in the SPCMD in which 1 is assigned to the SSLKP bit are different from the SSL signal output settings in the SPCMD to be used in the next transfer, this module switches the SSL signal status to SSL signal assertion ((5) in **Figure 24.16**) corresponding to the command for the next transfer. Notice that if such an SSL signal switching occurs, the slaves that drive the MISO signal compete, and the possibility arises of the collision of signal levels.

This module in master mode references within the module the SSL signal operation for the case where the SSLKP bit is not used. Even when the CPHA bit in SPCMD is 0, this module can accurately start serial transfers by asserting the SSL signal for the next transfer. For this reason, burst transfers in master mode can be executed irrespective of CPHA bit settings (see **Section 24.4.8(2), Slave Mode Operation**).

(e) RSPCK Delay (t1)

The RSPCK delay value in master mode depends on SCKDEN bit settings in the command register (SPCMD) and on clock delay register (SPCKD) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SCKDEN bit in the selected SPCMD and SPCKD, as shown in **Table 24.8**. For a definition of RSPCK delay, see **Section 24.4.4, Transfer Format**.

Table 24.8 Relationship among SCKDEN and SPCKD Settings and RSPCK Delay Values

SCKDEN	SPCKD	RSPCK Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(f) SSL Negation Delay (t2)

The SSL negation delay value in master mode depends on SLNDEN bit settings in the command register (SPCMD) and on SSL negation delay register (SSLND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SLNDEN bit in the selected SPCMD and SSLND, as shown in **Table 24.9**. For a definition of SSL negation delay, see **Section 24.4.4, Transfer Format**.

Table 24.9 Relationship among SLNDEN and SSLND Settings and SSL Negation Delay Values

SLNDEN	SSLND	SSL Negation Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(g) Next-Access Delay (t3)

The next-access delay value in master mode depends on SPNDEN bit settings in the command register (SPCMD) and on next-access delay register (SPND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPNDEN bit in the selected SPCMD and SPND, as shown in **Table 24.10**. For a definition of next-access delay, see **Section 24.4.4, Transfer Format**.

Table 24.10 Relationship among SPNDEN and SPND Settings and Next-Access Delay Values

SPNDEN	SPND	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 P0ϕ
1	000	1 RSPCK + 2 P0ϕ
	001	2 RSPCK + 2 P0ϕ
	010	3 RSPCK + 2 P0ϕ
	011	4 RSPCK + 2 P0ϕ
	100	5 RSPCK + 2 P0ϕ
	101	6 RSPCK + 2 P0ϕ
	110	7 RSPCK + 2 P0ϕ
	111	8 RSPCK + 2 P0ϕ

(h) Initialization Flowchart

Figure 24.17 is a flowchart illustrating an example of initialization in SPI operation when this module is used in master mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

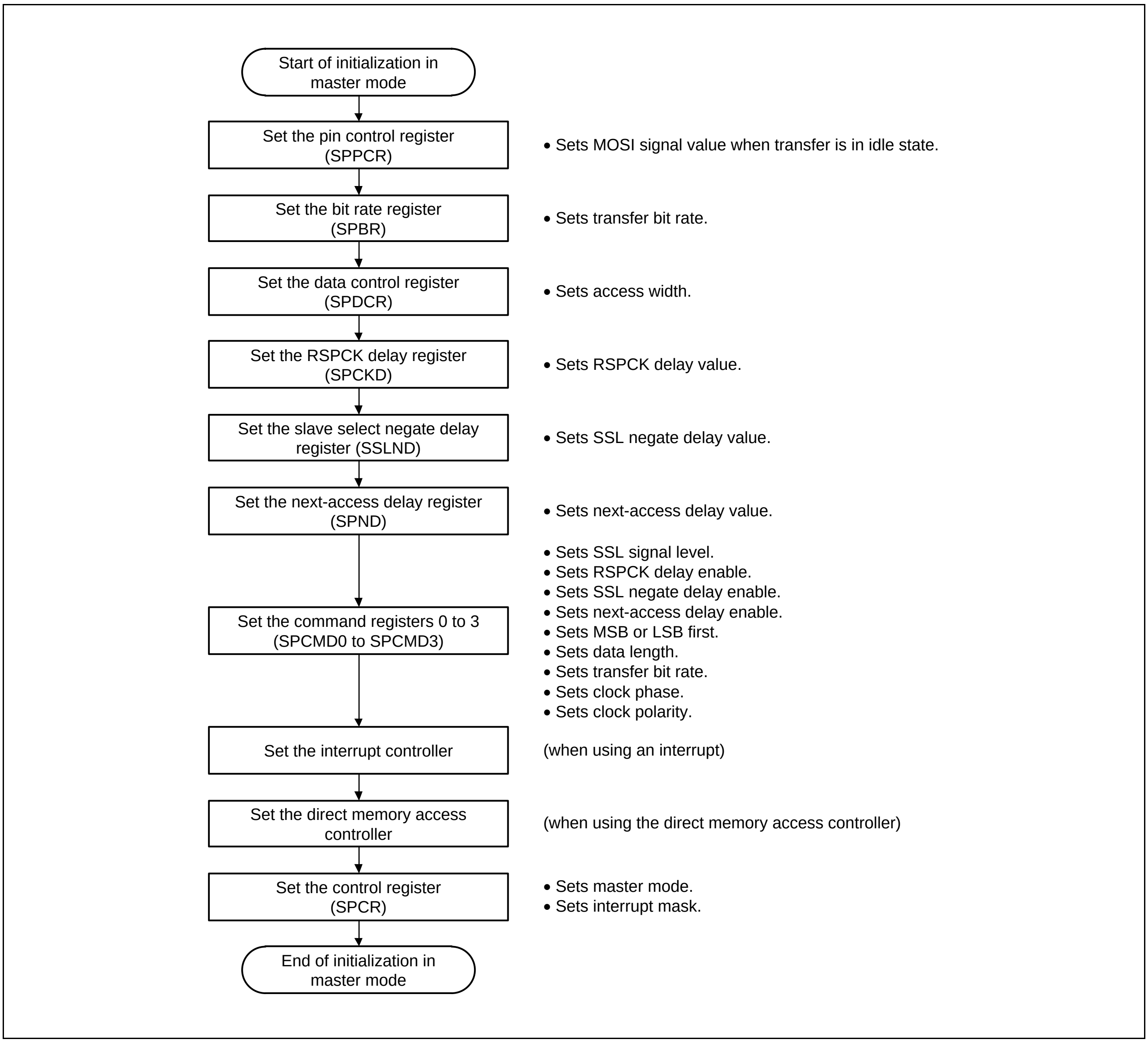


Figure 24.17 Example of Initialization Flowchart in Master Mode

(i) Transfer Operation Flowchart

Figure 24.18 is a flowchart illustrating a transfer in SPI operation when this module is used in master mode.

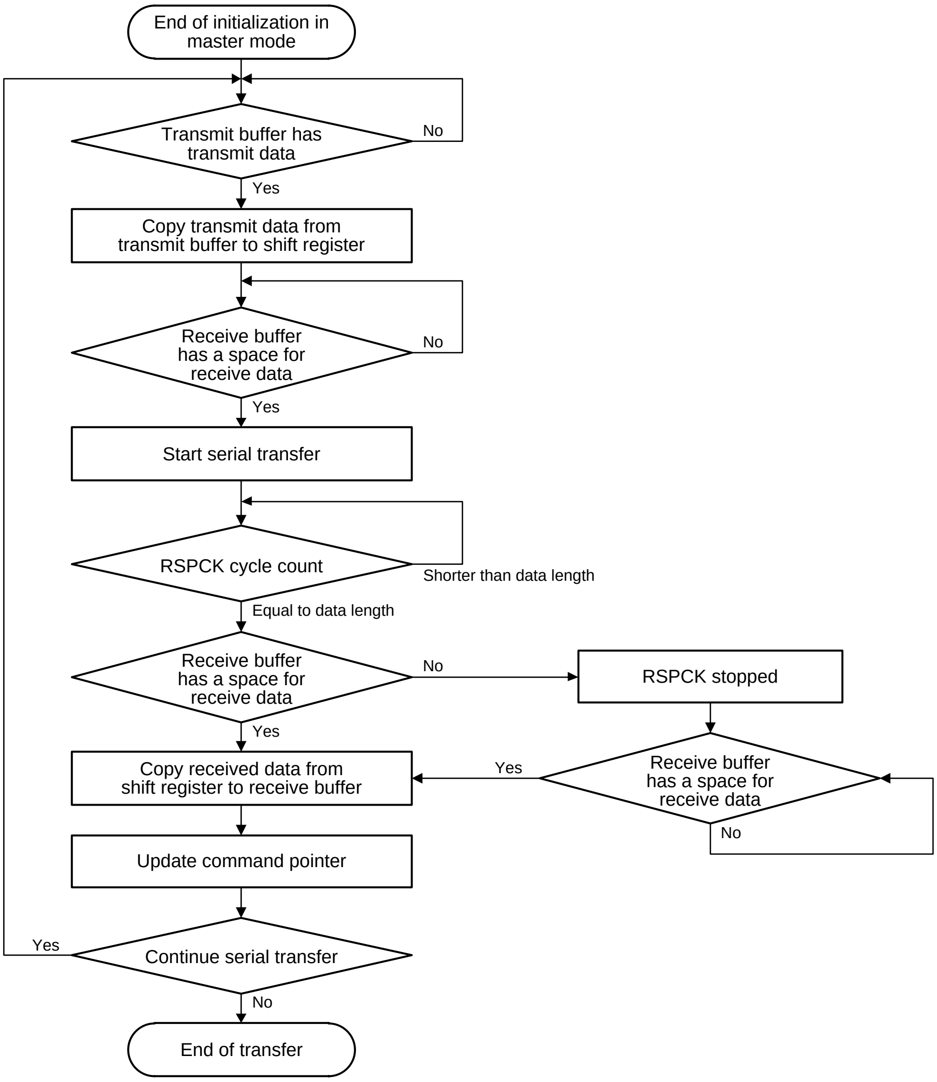


Figure 24.18 Transfer Operation Flowchart in Master Mode

(2) Slave Mode Operation

(a) Starting Serial Transfer

If this module detects an SSL input signal assertion when the CPHA bit in the command register 0 (SPCMD0) is 0, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 0, the asserting of the SSL input signal triggers the start of a serial transfer.

If this module detects the first RSPCK edge in an SSL signal asserted condition when the CPHA bit is 1, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 1, the first RSPCK edge in an SSL signal asserted condition triggers the start of a serial transfer.

When detecting the start of a serial transfer in a condition in which the shift register is empty, this module changes the status of the shift register to “full”, so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, this module leaves the status of the shift register intact, in the full state.

Irrespective of CPHA bit settings, this module starts driving MISO output signals at the SSL signal assertion timing. Whether the data output from this module is valid or invalid differs depending on CPHA bit settings.

For details on the transfer format, see **Section 24.4.4, Transfer Format**. The polarity of the SSL input signal depends on the setting of the SSL0P bit in the slave select polarity register (SSLP).

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register 0 (SPCMD0), this module terminates the serial transfer after detecting an RSPCK edge corresponding to the final sampling timing. When the receive buffer has an enough space for receive data, this module copies received data from the shift register to the receive buffer of the data register (SPDR) upon termination of the serial transfer. Irrespective of the value of the SPRF bit, this module changes the status of the shift register to “empty” upon termination of the serial transfer. If this module detects an SSL input signal negation from the beginning of serial transfer to the end of serial transfer, a mode fault error occurs (see **Section 24.4.6, Error Detection**).

The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the data length depends on the settings in bits SPB3 to SPB0 bits in SPCMD0. The polarity of the SSL input signal depends on the setting in the SSL0P bit in the slave select polarity register (SSLP). For details on the transfer format, see **Section 24.4.4, Transfer Format**.

(c) Notes on Slave Operations

If the CPHA bit in the command register 0(SPCMD0) is 0, this module starts serial transfers when it detects the assertion edge for an SSL input signal. In the type of configuration shown in **Figure 24.4** as an example, if this module is used in single-slave mode, the SSL signal is always fixed at active state. Therefore, when the CPHA bit is set to 0, this module cannot correctly start a serial transfer. To correctly execute send/receive operation in a configuration in which the SSL input signal is fixed at active state, the CPHA bit should be set to 1. When it is necessary to set the CPHA bit to 0, the SSL input signal should not be fixed.

(d) Burst Transfer

If the CPHA bit in the command register 0 (SPCMD0) is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSL input signal. If the CPHA bit is 1, the period from the first RSPCK edge to the sampling timing for the reception of the final bit in an SSL signal active state corresponds to a serial transfer period. Even when the SSL input signal remains at the active level, this module can accommodate burst transfers because it can detect the start of access.

If the CPHA bit is 0, for the reason given in **Section 24.4.8(2)(c), Notes on Slave Operations**, second and subsequent serial transfers during the burst transfer cannot be executed correctly.

(e) Initialization Flowchart

Figure 24.19 is a flowchart illustrating an example of initialization in SPI operation when this module is used in slave mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

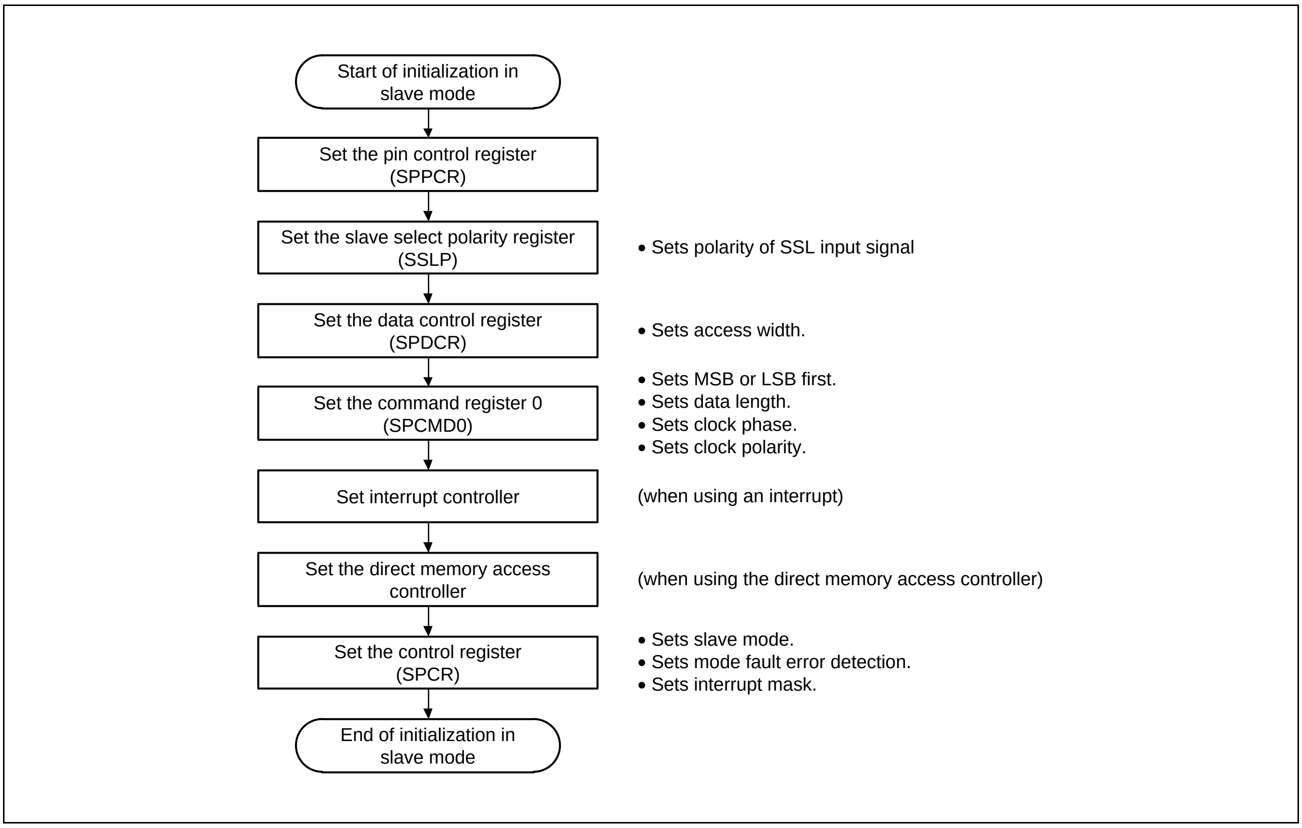


Figure 24.19 Example of Initialization Flowchart in Slave Mode

(f) Transfer Operation Flowchart (CPHA = 0)

Figure 24.20 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) set to 0.

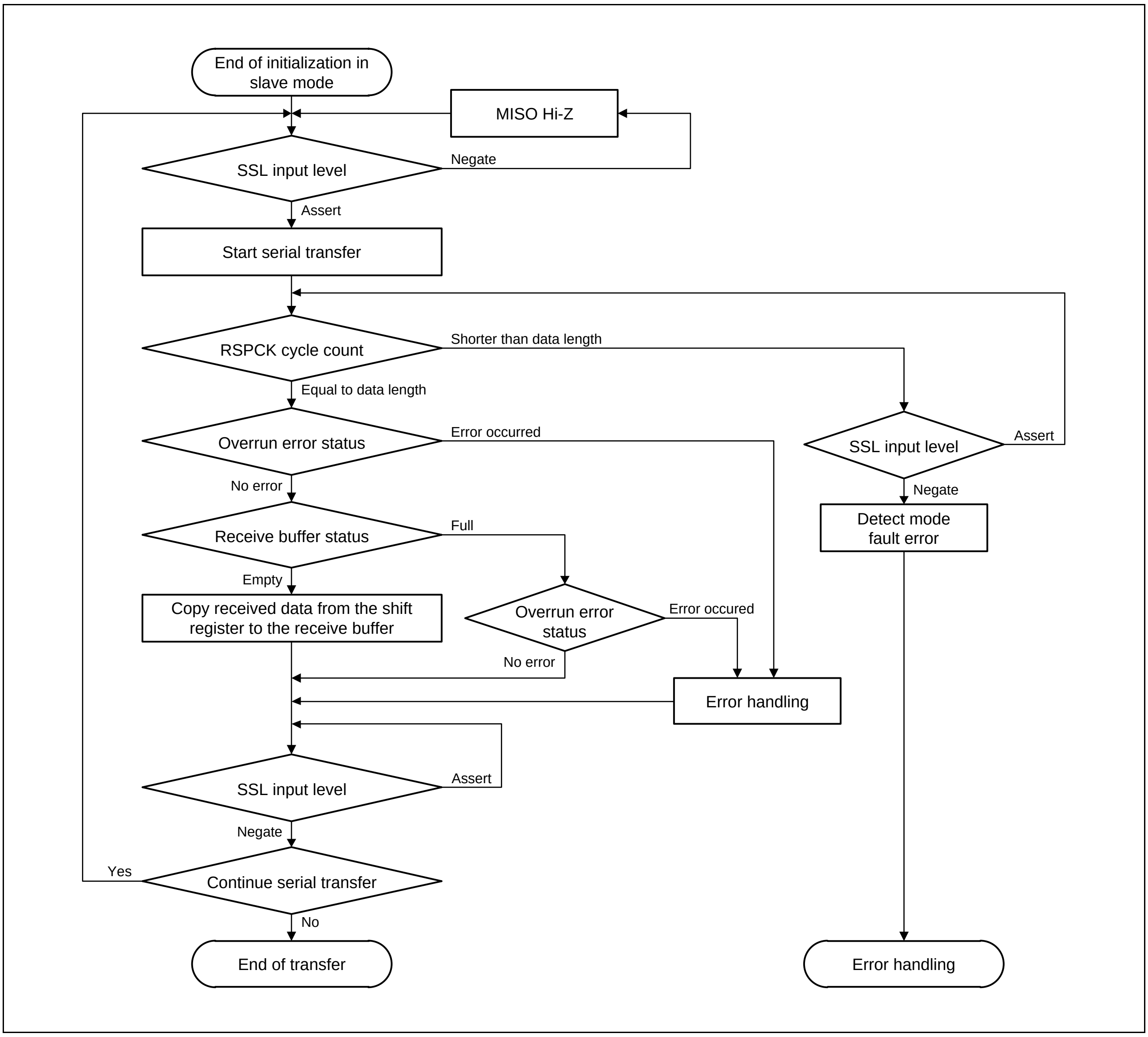


Figure 24.20 Transfer Operation Flowchart in Slave (CPHA = 0)

(g) Transfer Operation Flowchart (CPHA = 1)

Figure 24.21 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) and the MODFEN bit in the control register (SPCR) set to 1, respectively. The subsequent operation is not guaranteed when the serial transfer is started with the MODFEN bit set to 0 and the SSL input level is negated with the number of RSPCK cycles shorter than the data length.

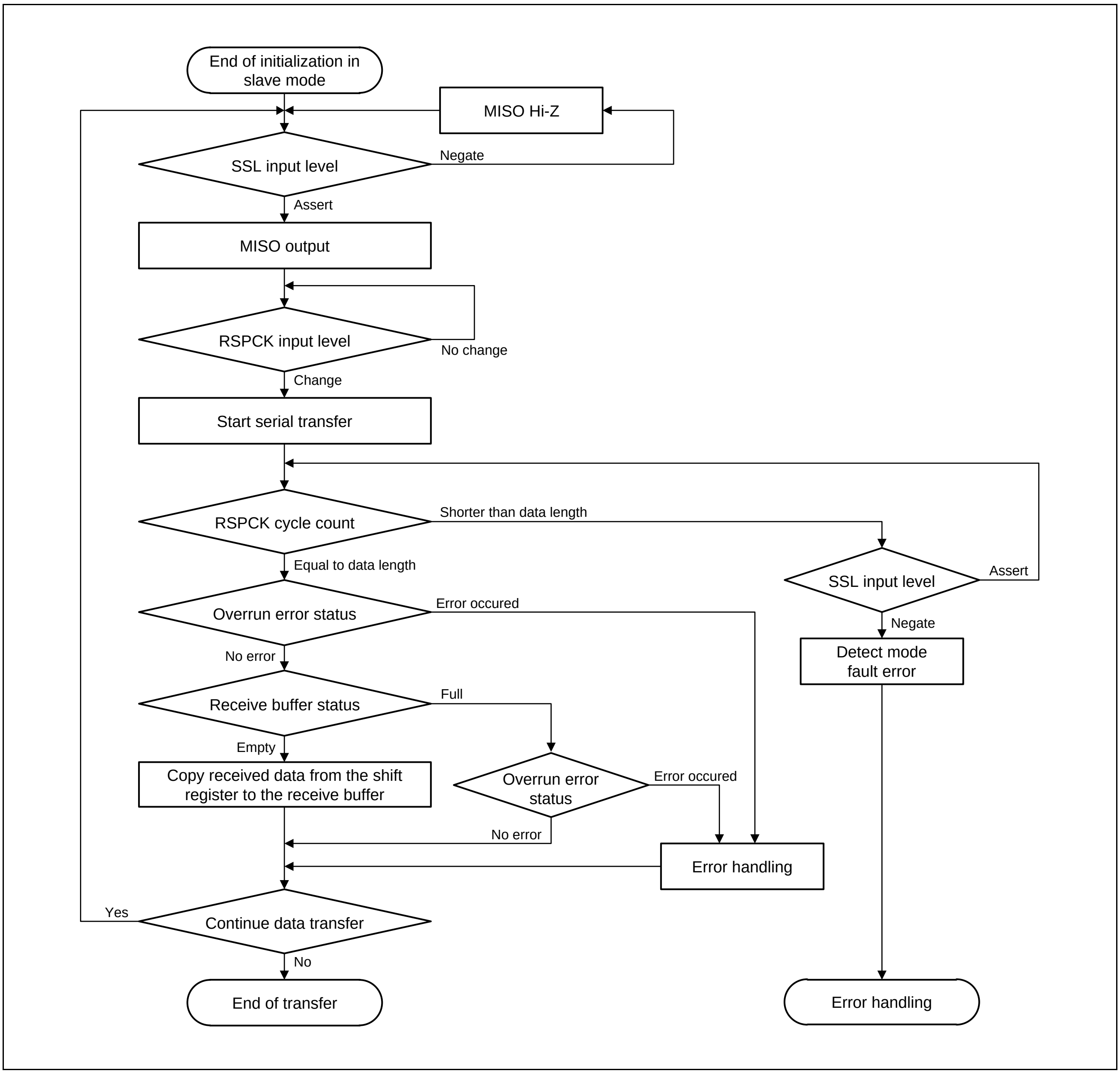


Figure 24.21 Transfer Operation Flowchart in Slave Mode (CPHA = 1)

24.4.9 Error Handling

Figure 24.22 and **Figure 24.23** show the error handling. The following error handling is used to return from the error state after an error in master or slave mode.

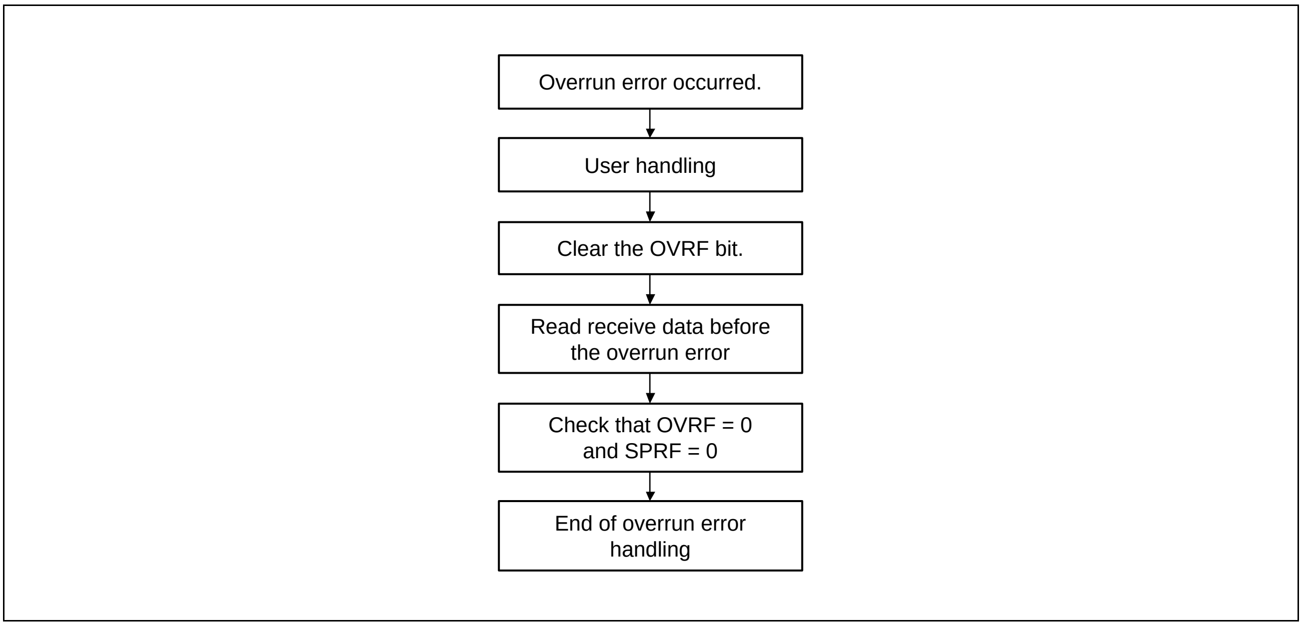


Figure 24.22 Error Handling (Overrun Error)

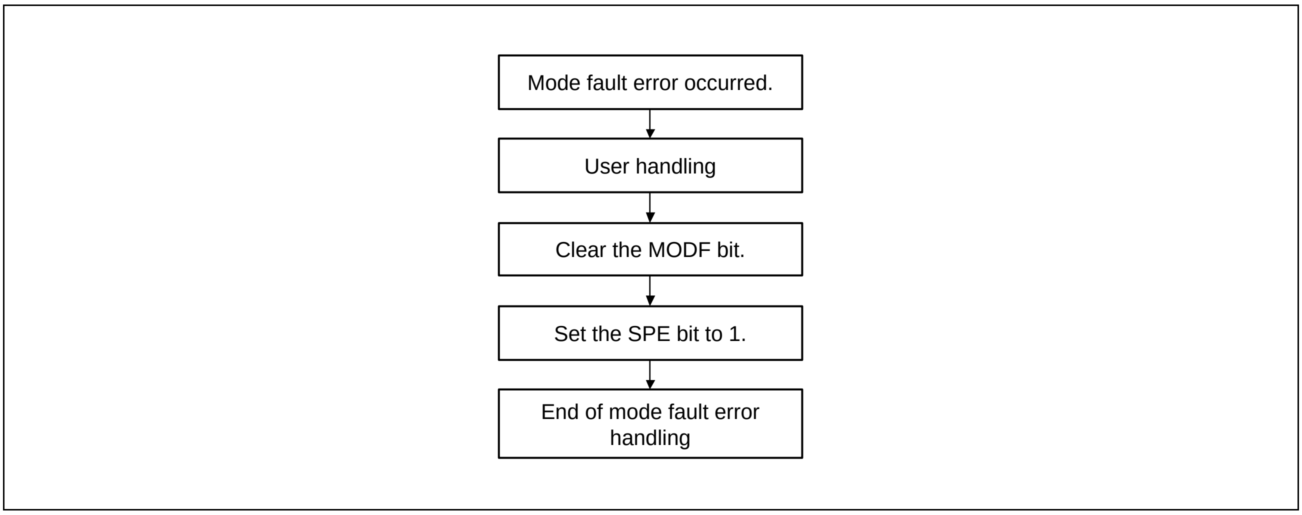


Figure 24.23 Error Handling (Mode Fault Error)

24.4.10 Loopback Mode

When 1 is written to the SPLP bit in the pin control register (SPPCR), this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects the input path and the output path (reversed) of the shift register. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data becomes the received data. **Figure 24.24** shows the configuration of the shift register input/output paths for the case where this module in master mode is set in loopback mode.

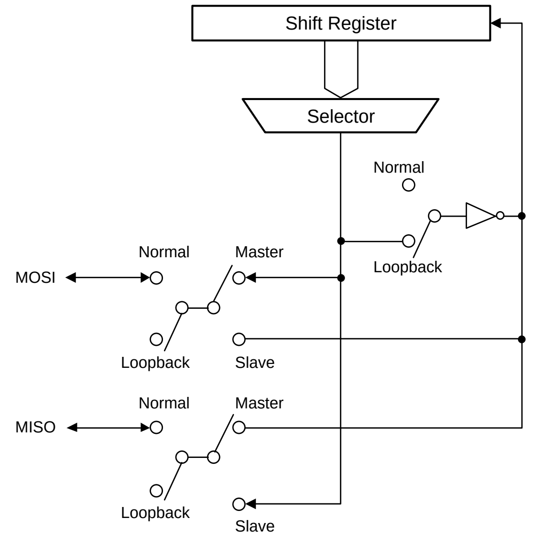


Figure 24.24 Configuration of Shift Register Input/Output Paths in Loopback Mode (Master Mode)

24.4.11 Interrupt Sources

This module has interrupt sources of receive buffer full, transmit buffer empty, mode fault, and overrun. In addition, the direct memory access controller can be activated by the receive buffer full or transmit buffer empty interrupt for data transfer.

Table 24.11 shows the interrupt sources.

When any of the interrupt conditions in **Table 24.11** is met, an interrupt is generated. The interrupt sources should be cleared with data transfer by the CPU or direct memory access controller.

Table 24.11 Interrupt Sources

Name	Interrupt Source	Abbreviation	Interrupt Condition	Activation of Direct Memory Access Controller
SPRI	Receive buffer full	RXI	(SPRIE = 1) • (SPRF = 1)	Possible
SPTI	Transmit buffer empty	TXI	(SPTIE = 1) • (SPTEF = 1)	Possible
SPEI	Mode fault	MOI	(SPEIE = 1) • (MODF = 1)	—
	Overrun	OVI	(SPEIE = 1) • (OVRF = 1)	—

25. SPI Multi I/O Bus Controller

The SPI multi I/O bus controller enables the direct connection of serial flash, OctaFlash™, or HyperFlash™ memory devices to this LSI chip.

25.1 Features

This module allows the connected serial flash, OctaFlash™, or HyperFlash™ memory devices to be accessed by reading the external address space, or using Manual mode to transmit and receive data.

25.1.1 Serial Flash Memory Interface

- SPI_PVDD: 1.8V/3.3V
- QSPI0_SPCLK: 66MHz (SDR), 50MHz (DDR)
- Up to two serial flash memory devices are connectable per channel.

  Note: Two serial flash memory devices are connected in parallel to implement an 8-bit bus interface.

- A data bus size of 1 bit or 4 bits can be selected for one serial flash memory device.
- Both single data rate (SDR) and double data rate (DDR) transfers are supported.
- Supports master operation. (Slave operation is not supported.)

25.1.2 OctaFlash™ Flash Memory Interface

- SPI_PVDD: 1.8V
- QSPI0_SPCLK: 100MHz
- One Octal-SPI flash memory device is connectable.

  Note: OctaFlash™ is a trademark of Macronix International Co., Ltd.

  Note: In this manual, an 8-bit SPI flash memory with 1 chip select, 1 clock source, and 1 data strobe configuration is called “Octal-SPI flash memory”.

25.1.3 HyperFlash^{TM} Interface

- SPI_PVDD: 1.8V
- QSPI0_SPCLK/QSPI1_SPCLK: 100MHz
- One HyperFlash memory device is connectable.
- The data bus is fixed to 8-bit width.
- Wrapped burst operation is not supported.

Note: HyperFlash is a trademark of Cypress Semiconductor Corporation.

25.1.4 External Address Space Read Mode

- Read access from the bus master to the SPI multi I/O space is automatically converted to a read command, and the read data is returned to the bus master.
- Normal read operation and burst read operation
- Efficient data reception due to built-in read cache (64-bit line × 32 entries)

25.1.5 Manual Mode

- Read and write commands are available for serial flash memory, Octal-SPI flash memory, or HyperFlash memory.
- A write buffer is provided to improve the efficiency of data write operations.
- The cache is not usable for reading.

25.2 Block Diagram

Figure 25.1 shows a block diagram of this module.

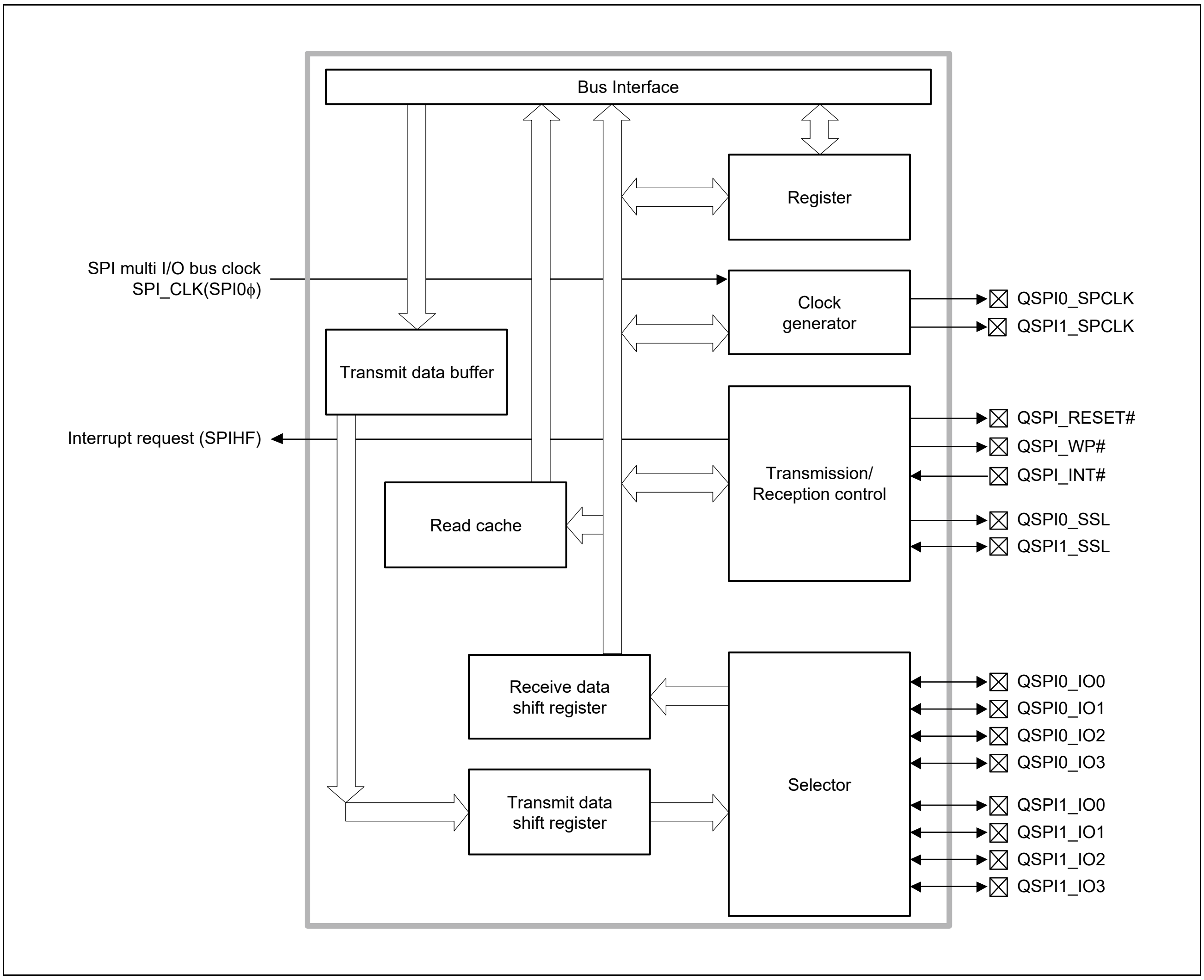


Figure 25.1 Block Diagram

25.3 Input/Output Pins

Table 25.1 shows the pin configuration for one channel.

The operating voltage on pins of the SPI multi I/O bus controller is 3.3 V or 1.8 V.

Before using the pins of the SPI multi I/O bus controller, be sure to set up the QSPI IO Voltage Control Register(QSPI) and Driving Ability Control Register (IOLH).

For details, see **Section 41.3.19, QSPI IO Voltage Control Register(QSPI)** and **Section 41.3.7, Driving Ability Control Register(IOLH)**.

Table 25.1 Pin Configuration

Pin Name	Symbol	I/O	One 4-bit Serial Flash connected. CMNCR.BSZ = 00	Two 4-bit Serial Flash connected. CMNCR.BSZ = 01	Octal-SPI flash memory connected. CMNCR.BSZ = 01	HyperFlash connected. CMNCR.BSZ = 01
Clock*[1]	QSPI0_SPCLK*[3]	Output	SCK	SCK	SCLK	CK
Data 0*[1]	QSPI0_IO0	I/O	SI/IO0	SI/IO0	SI/SIO0	DQ0
Data 1*[1]	QSPI0_IO1	I/O	SO/IO1	SO/IO1	SO/SIO1	DQ1
Data 2*[1]	QSPI0_IO2	I/O	WP#/IO2	WP#/IO2	SIO2	DQ2
Data 3*[1]	QSPI0_IO3	I/O	HOLD#/IO3	HOLD#/IO3	SIO3	DQ3
Slave select*[1]	QSPI0_SSL	Output	CS#	CS#	CS#	CS#
Clock	QSPI1_SPCLK*[3]	Output	—	SCK	—	CK#
Data 4	QSPI1_IO0	I/O	—	SI/IO0	SIO4	DQ4
Data 5	QSPI1_IO1	I/O	—	SO/IO1	SIO5	DQ5
Data 6	QSPI1_IO2	I/O	—	WP#/IO2	SIO6	DQ6
Data 7	QSPI1_IO3	I/O	—	HOLD#/IO3	SIO7	DQ7
Slave select	QSPI1_SSL	I/O	—	CS#	DQS (Data Strobe)	RWDS (Read Data Strobe)
Reset*[2]	QSPI_RESET#	Output	RESET#	RESET#	RESET#	RESET#
Write protect*[2]	QSPI_WP#	Output	For one 1-bit serial flash memory: WP#/IO2	—	—	—
Interrupt*[2]	QSPI_INT#	Input	—	—	—	INT#

Note 1. When connecting a serial flash memory device with the BSZ[1:0] bits set to 00, connect the serial flash memory to the QSPI0 pin group of this LSI.

Note 2. These pin connections must meet to the flash specifications.

Note 3. QSPIn_SPCLK (n = 0,1) is one-half frequency of SPI0ϕ.

25.4 Register Descriptions

Table 25.2 shows the register configuration.

Base Address: H’0_1006_0000 (Cortex-A55 Address Space)
Base Address: H’4006_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’5006_0000 (Cortex-M33 Address Space Secure)

Table 25.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common control register	CMNCR	R/W	H’01557301	H’0000	32
SSL delay register	SSLDR	R/W	H’00000000	H’0004	32
Data read control register	DRCR	R/W	H’001F0100	H’000C	32
Data read command setting register	DRCMR	R/W	H’00A00000	H’0010	32
Data read extended address setting register	DREAR	R/W	H’00000000	H’0014	32
Data read option setting register	DROPR	R/W	H’00000000	H’0018	32
Data read enable setting register	DRENR	R/W	H’A222D400	H’001C	32
Manual mode control register	SMCR	R/W	H’00000000	H’0020	32
Manual mode command setting register	SMCMR	R/W	H’00000000	H’0024	32
Manual mode address setting register	SMADR	R/W	H’00000000	H’0028	32
Manual mode option setting register	SMOPR	R/W	H’00000000	H’002C	32
Manual mode enable setting register	SMENR	R/W	H’00004000	H’0030	32
Manual mode read data register 0	SMRDR0	R	Undefined	H’0038	8, 16, 32
Manual mode read data register 1	SMRDR1	R	Undefined	H’003C	8, 16, 32
Manual mode write data register 0	SMWDR0	R/W	H’00000000	H’0040	8, 16, 32
Manual mode write data register 1	SMWDR1	R/W	H’00000000	H’0044	8, 16, 32
Common status register	CMNSR	R	H’00000001	H’0048	32
Data read dummy cycle setting register	DRDMCR	R/W	H’0000000B	H’0058	32
Data read DDR enable register	DRDRENR	R/W	H’00005101	H’005C	32
Manual mode dummy cycle setting register	SMDMCR	R/W	H’00000000	H’0060	32
Manual mode DDR enable register	SMDRENR	R/W	H’00000000	H’0064	32
PHY control register	PHYCNT	R/W	H’00000263	H’007C	32
PHY offset register 1	PHYOFFSET1	R/W	H’21511144	H’0080	32
PHY offset register 2	PHYOFFSET2	R/W	H’00000431	H’0084	32
PHY interrupt register	PHYINT	R/W	H’07070002	H’0088	32
PHY adjustment register 1	PHYADJ1	R/W	H’00000000	H’0070	32
PHY adjustment register 2	PHYADJ2	R/W	H’00000000	H’0074	32

Note: Do not write to any address those listed in the table. Otherwise, correct operation cannot be guaranteed. Reading from a non-listed address returns an undefined value.

25.4.1 Common Control Register (CMNCR)

CMNCR is a 32-bit register that controls the SPI multi I/O bus controller. The settings of this register are reflected both in external address space read mode and manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD	—	—	—	—	—	—	—	MOIIO3[1:0]		MOIIO2[1:0]		MOIIO1[1:0]		MOIIO0[1:0]	
Initial Value	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1
R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO3FV[1:0]		IO2FV[1:0]		—	—	IO0FV[1:0]		—	—	—	—	—	—	BSZ[1:0]	
Initial Value	0	1	1	1	0	0	1	1	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MD	0	R/W	Operating Mode Switch Switches the operating modes. 0: External address space read mode 1: Manual mode
30 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
23, 22	MOIIO3[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO3 Fixes the output value on QSPIn_IO3 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. *Note 1.* If an Octal-SPI flash memory device is connected, set these bits to 01. *Note 2.* If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).
21, 20	MOIIO2[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO2 Fixes the output value on QSPIn_IO2 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. *Note 1.* If an Octal-SPI flash memory device is connected, set these bits to 01. *Note 2.* If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).

Bit	Bit Name	Initial Value	R/W	Description
19, 18	MOIIO1[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO1 Fixes the output value on QSPIn_IO1 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. *Note 1.* If an Octal-SPI flash memory device is connected, set these bits to 01. *Note 2.* If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).
17, 16	MOIIO0[1:0]	01	R/W	QSPIn_SSL Output Idle Value Fix QSPIn_IO0 Fixes the output value on QSPIn_IO0 while QSPIn_SSL is negated. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state. *Note 1.* If an Octal-SPI flash memory device is connected, set these bits to 01. *Note 2.* If a HyperFlash memory device is connected, be sure to set these bits to 10 (the output value being that of the last bit in the previous transfer, with the pin placed in the Hi-Z state if that was the case in the previous transfer).
15, 14	IO3FV[1:0]	01	R/W	QSPIn_IO3 Fixed Value for 1-bit Size Fixes the output value of QSPIn_IO3 pin for 1-bit size. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state.
13, 12	IO2FV[1:0]	11	R/W	QSPIn_IO2 Fixed Value for 1-bit Size Fixes the output value of QSPIn_IO2 pin for 1-bit size. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	IO0FV[1:0]	11	R/W	QSPIn_IO0 Fixed Value for 1-bit Size Fixes the output value of QSPIn_IO0 pin for 1-bit size. 00: The output value is 0. 01: The output value is 1. 10: The output value is that of the last bit in the previous transfer (the pin is placed in the Hi-Z state if that was the case in the previous transfer). 11: The pin is placed in the Hi-Z state.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	BSZ[1:0]	01	R/W	Data Bus Size Specifies the number of serial flash memories to be connected. When HyperFlash connected, set to 01. 00: one serial flash memory connected 01: two serial flash memories connected, one HyperFlash connected, one Octal-SPI flash memory connected 1X: Setting prohibited *Note*: After changing (the value of) this bit field, all the entries in the read cache must be cleared by setting the RCF bit in DRCR to 1.

Note: The settings of the MOIIO3, MOIIO2, MOIIO1, and MOIIO0 bits for control over fixing the output values are reflected at the time QSPIn_SSL is negated on completion of data transfer.

25.4.2 SSL Delay Register (SSLDR)

SSLDR is a 32-bit register that adjusts the timing between the QSPIn_SSL signal and the QSPIn_SPCLK signal. The settings of this register are reflected both in external address space read mode and manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SPNDL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL[2:0]			—	—	—	—	—	SCKDL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	SPNDL[2:0]	000	R/W	Next Access Delay Sets the period from transfer end to next transfer start (next access). 000: 1 QSPIn_SPCLK (3 QSPIn_SPCLK when SSLE = 0) 001: 2 QSPIn_SPCLK 010: 3 QSPIn_SPCLK 011: 4 QSPIn_SPCLK 100: 5 QSPIn_SPCLK 101: 6 QSPIn_SPCLK 110: 7 QSPIn_SPCLK 111: 8 QSPIn_SPCLK
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	SLNDL[2:0]	000	R/W	QSPIn_SSL Negation Delay Sets the period from the time the last QSPIn_SPCLK edge is sent of a transfer to QSPIn_SSL pin negation (QSPIn_SSL negation delay). 000: 1 QSPIn_SPCLK 001: 2 QSPIn_SPCLK 010: 3 QSPIn_SPCLK 011: 4 QSPIn_SPCLK 100: 5 QSPIn_SPCLK 101: 6 QSPIn_SPCLK 110: 7 QSPIn_SPCLK 111: 8 QSPIn_SPCLK
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	SCKDL[2:0]	000	R/W	Clock Delay Sets the period from QSPIn_SSL pin assertion to QSPIn_SPCLK oscillation (clock delay). 000: 1.5 QSPIn_SPCLK 001: 2.5 QSPIn_SPCLK 010: 3.5 QSPIn_SPCLK 011: 4.5 QSPIn_SPCLK 100: 5.5 QSPIn_SPCLK 101: 6.5 QSPIn_SPCLK 110: 7.5 QSPIn_SPCLK 111: 8.5 QSPIn_SPCLK

25.4.3 Data Read Control Register (DRCR)

DRCR is a 32-bit register that sets the operation in external address space read mode.

The bits except the SSLN bit should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SSLN	—	—	—	RBURST[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RCF	RBE	—	—	—	—	—	—	—	SSLE
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SSLN	0	W	QSPIn_SSL Negation Asserted QSPIn_SSL can be negated by writing 1 to this bit when both the RBE and SSLE bits are 1. This bit is always read as 0. *Note*: To start next access after QSPIn_SSL negation using this bit, read SSLF in CMNSR = 0 to confirm that the QSPIn_SSL has been negated.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	RBURST[4:0]	11111	R/W	Read Data Burst Length Sets the burst length (data unit count) when reading. This bit field is enabled when the RBE bit is set to 1. 00000: 1 data unit 00001: 2 continuous data units : 11110: 31 continuous data units 11111: 32 continuous data units One data unit is 64 bits long.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	RCF	0	W	Read Cache Flush When 1 is written to this bit, all the entries in the read cache are cleared. This bit is always read as 0. After using the cache area as the write buffer, the cache area must be cleared by writing 1 to the RCF bit. *Note*: After flushing the read cache by writing 1 to the RCF bit, read the DRCR register before proceeding to read from the external address space.

Bit	Bit Name	Initial Value	R/W	Description
8	RBE	1	R/W	Read Burst Turns burst ON or OFF when reading. 0: Data is read according to the access size. 1: Read cache is enabled, and as many data units as the burst count specified in RBURST[4:0] bits is read. *Note*: When the cache is enabled and read access reaches the last address of the flash memory, the access address does not match the address held in the cache. Control the read address and access size so that read access does not reach the last address of the flash memory.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SSLE	0	R/W	QSPIn_SSL Negation Sets the conditions for QSPIn_SSL negation during read burst. QSPIn_SSL is negated for each access during normal read. 0: QSPIn_SSL is negated after transfer of data set in burst length. 1: QSPIn_SSL is negated when the accessed address is not continuous with the previously transferred address. *Note 1.* Set up this bit when a serial flash memory device is connected. *Note 2.* When this bit is set to 1, setting the RBURST[4:0] bits to H'01 is prohibited.

25.4.4 Data Read Command Setting Register (DRCMR)

DRCMR is a 32-bit register that sets the commands issued in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CMD[7:0]	H'A0	R/W	Command Sets the command.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	OCMD[7:0]	H'00	R/W	Optional Command Sets the optional command.

25.4.5 Data Read Extended Address Setting Register (DREAR)

DREAR is a 32-bit register that sets the address when the serial flash address is output in 32-bit mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	EAV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EAC[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	EAV[7:0]	H'00	R/W	32-Bit Extended Upper Address Fixed Value Sets the upper address bit values of the external address specified by the EAC[2:0] bits when the serial flash address is output in 32-bit mode. Bit 0 corresponds to the serial flash address bit [25], and bit 7 corresponds to the bit [32]. This setting is valid when the ADE[3] bit in DRENR is 1. When EAC[2:0] are 000, serial flash address [32:25] fixed values should set to EAV[7:0]. When EAC[2:0] are 001, serial flash address [32:26] fixed values should set to EAV[7:1]. When EAC[2:0] are 010, serial flash address [32:27] fixed values should set to EAV[7:2]. When EAC[2:0] are 011, serial flash address [32:28] fixed values should set to EAV[7:3]. (1) When BSZ[1:0] in CMNCR = 00 (one serial flash memory connected) Serial flash addresses [31:0] are used for accessing. (2) When BSZ[1:0] in CMNCR = 01 (two serial flash memories connected) Serial flash addresses [32:1] are used for accessing.
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	EAC[2:0]	000	R/W	32-Bit Extended External Address Valid Range Sets the range of the external address to be used as serial flash address when the serial flash address is output in 32-bit mode. This setting is valid when the ADE[3] bit in DRENR is 1. 000: External address bits [24:0] enabled 001: External address bits [25:0] enabled 010: External address bits [26:0] enabled 011: External address bits [27:0] enabled Other than above: Setting prohibited

25.4.6 Data Read Option Setting Register (DROPR)

DROPR is a 32-bit register that sets the option data in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPD3[7:0]								OPD2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OPD1[7:0]								OPD0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OPD3[7:0]	H’00	R/W	Option Data 3 Sets the option data 3.
23 to 16	OPD2[7:0]	H’00	R/W	Option Data 2 Sets the option data 2.
15 to 8	OPD1[7:0]	H’00	R/W	Option Data 1 Sets the option data 1.
7 to 0	OPD0[7:0]	H’00	R/W	Option Data 0 Sets the option data 0.

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

25.4.7 Data Read Enable Setting Register (DRENR)

DRENR is a 32-bit register that sets the bit size of the command, optional command, address, option data, and read data in external address space read mode and enables outputting them other than read data.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

When a HyperFlash or Octal-SPI flash memory device is to be connected, refer to **Table 25.6, Enable Register (HyperFlash)**, or **Table 25.7, Enable Register (Octal-SPI Flash Memory)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	DRDB[1:0]	
Initial Value	1	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				—	—	—	—
Initial Value	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	CDB[1:0]	10	R/W	Command Bit Size Sets the command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
29, 28	OCDB[1:0]	10	R/W	Optional Command Bit Size Sets the optional command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	ADB[1:0]	10	R/W	Address Bit Size Sets the address size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	OPDB[1:0]	10	R/W	Option Data Bit Size Sets the option data size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
17, 16	DRDB[1:0]	10	R/W	Data Read Bit Size Sets the data read size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
15	DME	1	R/W	Dummy Cycle Enable Enables insertion of the dummy cycle before the read data. *Note*: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled
14	CDE	1	R/W	Command Enable Sets whether or not the command is to be output. 0: Command output disabled 1: Command output enabled *Note*: When an Octal-SPI flash memory device is connected and this bit set to 1, set the CDE bit in SMENR register to 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	OCDE	1	R/W	Optional Command Enable Sets whether or not the optional command is to be output. 0: Optional command output disabled 1: Optional command output enabled *Note*: When an Octal-SPI flash memory device is connected, set this bit to 0.
11 to 8	ADE[3:0]	0100	R/W	Address Enable Sets the address to be output. Be sure to use the following setting; otherwise, the operation is not guaranteed. (1) When a serial flash memory device is connected 0000: Output disabled 0111: Address[23:0] 1111: Address[31:0] Other than above: Setting prohibited (2) When two serial flash memory devices or one Octal-SPI flash memory device is connected 0000: Output disabled 0111: Address[24:1] 1111: Address[32:1] 1100: Octal-SPI flash memory addresses (operation is in accord with the 8-8-8 protocol in **Table 25.15**) Other than above: Setting prohibited (3) When a HyperFlash memory device is connected 0100: HyperFlash addresses Other than above: Setting prohibited
7 to 4	OPDE[3:0]	0000	R/W	Option Data Enable Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

25.4.8 Manual Mode Control Register (SMCR)

SMCR is a 32-bit register that sets the operation in Manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSLKP	—	—	—	—	—	SPIRE	SPIWE	SPIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SSLKP	0	R/W	QSPIn_SSL Signal Level Determines the QSPIn_SSL status after the end of transfer. 0: QSPIn_SSL signal is negated at the end of transfer. 1: QSPIn_SSL signal level is maintained from the end of transfer to the start of next access. *Note*: The setting to start data transfer for reading is prohibited while the setting of SSLKP is 1.
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SPIRE	0	R/W	Data Read Enable Enables reading in manual mode. 0: Data reading disabled 1: Data reading enabled *Note*: When the transfer data bit size is set to 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.
1	SPIWE	0	R/W	Data Write Enable Enables writing in manual mode. 0: Data writing disabled 1: Data writing enabled *Note*: When the transfer data bit size is set to 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.
0	SPIE	0	W	SPI Data Transfer Enable Data is transferred by setting this bit to 1. This bit is enabled only when the TEND bit in CMNSR is set to 1. The operation cannot be guaranteed when this bit is set to 1 with the TEND bit set to 0. This bit is always read as 0. *Note*: When the QSPIn_SSL pin is de-asserted, the command, optional command, address, and option data that are output enabled are output even if the SPIRE and SPIWE bits are set to 0. When the QSPIn_SSL pin is asserted, follow the notes described in **Section 25.6.3, Notes on Starting Transfer from the QSPIn_SSL Retained State in Manual Mode**.

25.4.9 Manual Mode Command Setting Register (SMCMR)

SMCMR is a 32-bit register that sets the commands issued in Manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CMD[7:0]	H’00	R/W	Command Sets the command.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	OCMD[7:0]	H’00	R/W	Optional Command Sets the optional command.

25.4.10 Manual Mode Address Setting Register (SMADR)

SMADR is a 32-bit register that sets the addresses in Manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR[31:24]								ADR[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADR[31:24]	H’00	R/W	Address Sets the value of bits 31 to 24 when the serial flash address is output in 32-bit units. This setting is valid when ADE[3] in SMENR is 1.
23 to 0	ADR[23:0]	H’000000	R/W	Address Sets the address.

25.4.11 Manual Mode Option Setting Register (SMOPR)

SMOPR is a 32-bit register that sets the option data in Manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPD3[7:0]								OPD2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OPD1[7:0]								OPD0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OPD3[7:0]	H'00	R/W	Option Data 3 Sets the option data 3.
23 to 16	OPD2[7:0]	H'00	R/W	Option Data 2 Sets the option data 2.
15 to 8	OPD1[7:0]	H'00	R/W	Option Data 1 Sets the option data 1.
7 to 0	OPD0[7:0]	H'00	R/W	Option Data 0 Sets the option data 0.

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

25.4.12 Manual Mode Enable Setting Register (SMENR)

SMENR is a 32-bit register that sets the bit size of the command, optional command, address, option data, and transfer data in Manual mode and enables their output. SMENR also enables dummy cycle insertion. Disabling all of the command, optional command, address, option data, dummy cycle, and transfer data is prohibited. At least one of them except dummy cycle must be enabled.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

When a HyperFlash or Octal-SPI flash memory device is to be connected, refer to **Table 25.6, Enable Register (HyperFlash)**, or **Table 25.7, Enable Register (Octal-SPI Flash Memory)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	SPIDB[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				SPIDE[3:0]			
Initial Value	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	CDB[1:0]	00	R/W	Command Bit Size Sets the command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
29, 28	OCDB[1:0]	00	R/W	Optional Command Bit Size Sets the optional command size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	ADB[1:0]	00	R/W	Address Bit Size Sets the address size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	OPDB[1:0]	00	R/W	Option Data Bit Size Sets the option data size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
17, 16	SPIDB[1:0]	00	R/W	Transfer Data Bit Size Sets the transfer data size in bit units. 00: 1 bit 10: 4 bits Other than above: Setting prohibited
15	DME	0	R/W	Dummy Cycle Enable Enables insertion of the dummy cycle before the read data. *Note*: Dummy cycle insertion is prohibited for write in Manual mode including the case in which a transfer ends with a dummy cycle. *Note*: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled
14	CDE	1	R/W	Command Enable Sets the command to be output. 0: Command output disabled 1: Command output enabled *Note*: When an Octal-SPI flash memory device is connected and the CDE bit in DRENR register set to 1, set this bit to 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	OCDE	0	R/W	Optional Command Enable Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled *Note*: When an Octal-SPI flash memory device is connected, set this bit to 0.
11 to 8	ADE[3:0]	0000	R/W	Address Enable Sets the address to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. (1) When a serial flash memory device or an Octal-SPI flash memory device is connected 0000: Output disabled 0100: ADR[23:16] 0110: ADR[23:8] 0111: ADR[23:0] 1111: ADR[31:0] 1100: Octal-SPI flash memory (operation is in accord with the 8-8-8 protocol in **Table 25.15**) Other than above: Setting prohibited (2) When a HyperFlash memory device is connected 0100: HyperFlash memory *Note*: The setting of these bits is required to be 0111 or 1111 when the write buffer is in use.
7 to 4	OPDE[3:0]	0000	R/W	Option Data Enable Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	SPIDE[3:0]	0000	R/W	Transfer Data Enable Sets valid transfer data. Valid data differs depending on the BSZ[1:0] bit setting in CMNCR. The following settings must be used. Otherwise, the operation is not guaranteed. (1) BSZ[1:0] bits in CMNCR = 00 (one serial flash memory connected) 0000: Not transferred 1000: 8 bits transferred (enables data at address 0 of the Manual mode read/write data registers 0) 1100: 16 bits transferred (enables data at addresses 0 and 1 of the Manual mode read/write data registers 0) 1111: 32 bits transferred (enables data at addresses 0 to 3 of the Manual mode read/write data registers 0) Other than above: Setting prohibited (2) BSZ[1:0] bits in CMNCR = 01 (two serial flash memories connected) 0000: Not transferred 1000: 16 bits transferred (enables data at addresses 0 and 1 of the Manual mode read/write data registers 0) 1100: 32 bits transferred (enables data at addresses 0 to 3 of the Manual mode read/write data registers 0) 1111: 64 bits transferred (enables data at addresses 0 to 3 of the Manual mode read/write data registers 0 and data at addresses 0 to 3 of the Manual mode read/write data registers 1) Other than above: Setting prohibited

25.4.13 Manual Mode Read Data Register 0 (SMRDR0)

SMRDR0 is a 32-bit register that stores the read data in Manual mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from LSB.

The settings of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDATA0[31:16]															
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDATA0[15:0]															
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDATA0[31:0]	Undefined	R	Read Data Holds the data read in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Read data[31:0]. BSZ[1:0] = 01: Read data[63:32].

Note: The contents of this register and SMRDR1 are modified upon completion of reception in manual mode. Be sure to read data when reception in manual mode is completed.

25.4.14 Manual Mode Read Data Register 1 (SMRDR1)

SMRDR1 is a 32-bit register that stores the read data in Manual mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from LSB.

The settings of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDATA1[31:16]															
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDATA1[15:0]															
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDATA1[31:0]	Undefined	R	Read Data Holds the data read in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Bits in this register are disabled. BSZ[1:0] = 01: Read data[31:0]

25.4.15 Manual Mode Write Data Register 0 (SMWDR0)

SMWDR0 is a 32-bit register that sets the write data in Manual mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDATA0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDATA0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	WDATA0[31:0]	All 0	R/W	Write Data Holds the data to be written in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Write data[31:0]. BSZ[1:0] = 01: Write data[63:32]. *Note*: When the Octal-SPI flash memory is connected (OPI), the write data should be arranged and written as follows. ● To write data D[31:0] in the STR mode WDATA0[31:0] = {D[23:20], D[31:28], D[19:16], D[27:24], D[7:4], D[15:12], D[3:0], D[11:8]} ● To write data D[31:0] in the DTR mode WDATA0[31:0] = {D[31:28], D[23:20], D[27:24], D[19:16], D[15:12], D[7:4], D[11:8], D[3:0]}

25.4.16 Manual Mode Write Data Register 1 (SMWDR1)

SMWDR1 is a 32-bit register that sets the write data in Manual mode.

This register is enabled when the BSZ[1:0] bits in CMNCR are set to 01 and disabled when the BSZ[1:0] bits in CMNCR are set to 00.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the Manual mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDATA1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDATA1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	WDATA1[31:0]	All 0	R/W	Write Data Holds the data to be written in Manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Bits in this register are disabled. BSZ[1:0] = 01: Write data[31:0]. *Note*: When the Octal-SPI flash memory is connected (OPI), the write data should be arranged and written as follows. ● To write data D[31:0] in the STR mode WDATA1[31:0]= {D[23:20], D[31:28], D[19:16], D[27:24], D[7:4], D[15:12], D[3:0], D[11:8]} ● To write data D[31:0] in the DTR mode WDATA1[31:0] = {D[31:28], D[23:20], D[27:24], D[19:16], D[15:12], D[7:4], D[11:8], D[3:0]}

25.4.17 Common Status Register (CMNSR)

CMNSR is a 32-bit register that holds flags indicating the operating state.

The settings of this register are reflected both in external address space read mode and Manual mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSLF	TEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SSLF	0	R	QSPIn_SSL Pin Monitor 0: QSPIn_SSL pin is negated 1: QSPIn_SSL pin is asserted
0	TEND	1	R	Transfer End Flag Indicates whether the data transfer has ended. 0: Indicates that data transfer is in progress 1: Indicates that data transfer has ended

25.4.18 Data Read Dummy Cycle Setting Register (DRDMCR)

DRDMCR is a 32-bit register that sets the number of dummy cycles to be inserted in external address space read mode. The settings of this register are enabled when the DME bit in the data read enable setting register (DRENR) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DMCYC[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	DMCYC[4:0]	H'0B	R/W	Number of Dummy Cycles Sets the number of dummy cycles to be inserted when the DME bit in the data read enable setting register (DRENR) is 1. 00001: 2 cycles 00010: 3 cycles … 10010: 19 cycles 10011: 20 cycles Other than above: Setting prohibited

25.4.19 Data Read DDR Enable Register (DRDRENR)

DRDRENR is a 32-bit register that sets SDR or DDR transfer of the address, option data, and read data in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	HYPE			—	—	—	ADDRE	—	—	—	OPDRE	—	—	—	DRDRE
Initial Value	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	HYPE	101	R/W	HyperFlash, Octal-SPI flash memory DDR mode enable 101: HyperFlash, Octal-SPI flash memory DDR mode 100: Only commands are issued in DDR mode when an Octal-SPI flash memory device is connected. 000: SPI flash mode Other than above: Setting prohibited
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ADDRE	1	R/W	Address DDR Enable Sets SDR or DDR transfer of the address. 0: SDR transfer 1: DDR transfer *Note*: Set this bit to 1 when the HyperFlash memory is connected.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OPDRE	0	R/W	Option Data DDR Enable Sets SDR or DDR transfer of the option data. 0: SDR transfer 1: DDR transfer
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DRDRE	1	R/W	Data Read DDR Enable Sets SDR or DDR transfer of the read data. 0: SDR transfer 1: DDR transfer

25.4.20 Manual Mode Dummy Cycle Setting Register (SMDMCR)

SMDMCR is a 32-bit register that sets the number of dummy cycles to be inserted in Manual mode.

The settings of this register are enabled when the DME bit in the Manual mode enable setting register (SMENR) is 1. The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DMCYC[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	DMCYC[4:0]	00000	R/W	Number of Dummy Cycles Sets the number of dummy cycles to be inserted when the DME bit in the Manual mode enable setting register (SMENR) is 1. 00001: 2 cycles 00010: 3 cycles .... 10010: 19 cycles 10011: 20 cycles Other than above: Setting prohibited

25.4.21 Manual Mode DDR Enable Register (SMDRENR)

SMDRENR is a 32-bit register that sets SDR or DDR transfer of the address, option data, and transfer data in Manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	HYPE[2:0]			—	—	—	ADDRE	—	—	—	OPDRE	—	—	—	SPIDRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	HYPE[2:0]	000	R/W	HyperFlash, Octal-SPI flash memory DDR mode enable 101: HyperFlash, Octal-SPI flash memory DDR mode 100: Only commands are issued in DDR mode when an Octal-SPI flash memory device is connected. 000: SPI flash mode Other than above: Setting prohibited
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ADDRE	0	R/W	Address DDR Enable Sets SDR or DDR transfer of the address. 0: SDR transfer 1: DDR transfer *Note*: Set this bit to 1 when the HyperFlash memory is connected.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OPDRE	0	R/W	Option Data DDR Enable Sets SDR or DDR transfer of the option data. 0: SDR transfer 1: DDR transfer
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SPIDRE	0	R/W	Transfer Data DDR Enable Sets SDR or DDR transfer of the transfer data. 0: SDR transfer 1: DDR transfer *Note*: Set this bit to 1 when the HyperFlash memory is connected.

25.4.22 PHY Control Register (PHYCNT)

PHYCNT is a 32-bit register that sets the PHY operation mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAL	ALT_ALIGN	—	—	—	—	—	—	OCTA[1:0]		EXDS	OCT	—	HS	CKSEL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	WBUF2	—	WBUF	PHYMEM[1:0]	
Initial Value	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CAL	0	W	PHY Calibration Specifies whether to perform PHY calibration. Be sure to perform calibration before access in Manual mode. 0: Calibration is not performed. 1: Calibration is performed.
30	ALT_ALIGN	0	R/W	Octal-SPI flash memory Alternative Alignment 0: Alternative alignment for Octal-SPI flash memory connection is not supported. 1: Alternative alignment for Octal-SPI flash memory connection is supported.
29 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23, 22	OCTA[1:0]	00	R/W	Octal-SPI flash memory Alignment Specifies the data alignment in the Octal-SPI flash memory in DDR mode. For details, refer to **Section 25.5.14, Data Alignment for Octal-SPI Flash Memory**. When the connected memory is HyperFlash or serial flash memory, set these bits to 00. 00: Specify this value when the HyperFlash, Octal-SPI flash memory (in SDR mode), or serial flash memory is connected. 01: Alternative alignment is supported. 10: Sequential alignment is supported. 11: Setting prohibited. *Note 1.* When using alternative alignment with Octal-SPI flash memory, also set the ALT_ALIGN bit to 1. *Note 2.* When using sequential alignment with Octal-SPI flash memory, the setting of the ALT_ALIGN bit is treated as “don’t care”.
21	EXDS	0	R/W	External Data Strobe When the data strobe line is connected to the serial flash memory, the external data strobe signal sent from the serial flash memory is used. When the Octal-SPI flash memory is connected, set this bit to 1. *Note*: When the HyperFlash memory is connected, set this bit to 0. 0: The external data strobe signal is not used. 1: The external data strobe signal is used.
20	OCT	0	R/W	Octal-SPI flash memory Protocol Mode 0: A mode other than the Octa protocol mode is used. 1: Octal-SPI flash memory protocol mode is used.

Bit	Bit Name	Initial Value	R/W	Description
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	HS	0	R/W	High-Speed Response Mode Specifies the high-speed response mode. 0: Data is read for the number of data units specified in the RBURST bits of the DRCR register and then output to the bus master. 1: The read data is output to bus master in parallel of device access when DRCR.RBE = 1. *Note 1.* When this bit is set to 1, use DMA transfer. The transfer size in the RBURST[4:0] bits of the DRCR register should be fixed to H'1F. *Note 2.* Do not access the register area during DMA transfer.
17, 16	CKSEL[1:0]	00	R/W	Clock Timing Switching Adjusts the timing of the clock when serial flash is connected. Always follow the procedure for setting shown in **Section 25.5.17, Timing Adjustment**. Be sure to set the CKSEL[1:0] bits to 11b when Octal-SPI flash memory or HyperFlash is connected.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
8, 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6, 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	WBUF2	0	R/W	Write Buffer Enable 2 Specifies whether to use the write buffer to write data to the flash memory. For the usage of the write buffer, refer to **Section 25.5.13, Write Buffer Operation**. 0: The write buffer is not used. 1: The write buffer is used to write data to the flash memory.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	WBUF	0	R/W	Write Buffer Enable Specifies whether to use the write buffer to write data to the flash memory. For the usage of the write buffer, refer to **Section 25.5.13, Write Buffer Operation**. 0: The write buffer is not used. 1: The write buffer is used to write data to the flash memory.
1, 0	PHYMEM[1:0]	11	R/W	Device Selection Selects the device to be connected. 00: Serial flash or Octal-SPI flash memory in SDR mode 01: Serial flash or Octal-SPI flash memory in DDR mode 11: HyperFlash memory Other than above: Setting prohibited

25.4.23 PHY Offset Register 1 (PHYOFFSET1)

PHYOFFSET1 is a 32-bit register that sets the timing adjustment in the DDR operation.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DDRTMG		—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	1	0	0	0	0	1	0	1	0	1	0	0	0	1
R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29, 28	DDRTMG	10	R/W	DDR Operation Timing Adjusts the timing of reading in the DDR mode. 10: Specify this value for reading when SMDRENR.SPIDRE = 1 or DRDRENR.DRDRE = 1. 11: Specify this value for reading when SMDRENR.SPIDRE = 0 or DRDRENR.DRDRE = 0. Other than above: Setting prohibited
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

25.4.24 PHY Offset Register 2 (PHYOFFSET2)

PHYOFFSET2 is a 32-bit register that sets the timing adjustment in the DDR operation.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OCTTMG			—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	OCTTMG	100	R/W	Octal-SPI flash memory Operation Timing Adjusts the timing of writing to the Octal-SPI flash memory 000: Specify this value when using the serial flash memory with write buffer operation of the bit width 1-1-4 or 1-4-4 in **Table 25.14**. 100: Specify this value when the serial flash or HyperFlash memory is used. 011: Specify this value when the Octal-SPI flash memory is used. Other than above: Setting prohibited *Note*: When the Octal-SPI flash memory is used in the serial flash compatible mode, set these bits to 100.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

25.4.25 PHY Interrupt Register (PHYINT)

PHYINT is a 32-bit register for making settings to do with interrupt signals and pins.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RSTEN	WPEN	INTEN	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RSTVAL	WPVAL	INT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	RSTEN	1	R/W	QSPI_RESET# Pin Enable Enables or disables the QSPI_RESET# pin. 0: The QSPI_RESET# pin is disabled. 1: The QSPI_RESET# pin is enabled and the value specified in the RSTVAL bit is output.
25	WPEN	1	R/W	QSPI_WP# Pin Enable Enables or disables the QSPI_WP# pin. 0: The QSPI_WP# pin is disabled. 1: The QSPI_WP# pin is enabled and the value specified in the WPVAL bit is output.
24	INTEN	1	R/W	QSPI_INT# Pin Enable Enables or disables the QSPI_INT# pin. 0: The QSPI_INT# pin is disabled. 1: The QSPI_INT# pin is enabled and the interrupt signal sent from the HyperFlash memory is enabled.
23 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	RSTVAL	0	R/W	Value Output through QSPI_RESET# Pin Specifies the value to be output through the QSPI_RESET# pin. This setting takes effect when the RSTEN bit is set to 1. 0: QSPI_RESET# = H 1: QSPI_RESET# = L
1	WPVAL	1	R/W	Value Output through QSPI_WP# Pin Specifies the value to be output through the QSPI_WP# pin. This setting takes effect when the WPEN bit is set to 1. 0: QSPI_WP# = H 1: QSPI_WP# = L

Bit	Bit Name	Initial Value	R/W	Description
0	INT	0	R	Interrupt Status When the QSPI_INT# signal goes low, this bit is set to 1 to indicate the occurrence of an interrupt in the connected device.

25.4.26 PHY Adjustment Register 1 (PHYADJ1)

PHYADJ1 is a 32-bit register used to adjust the timing when serial flash memory is connected.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADJ1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADJ1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADJ1[31:0]	All 0	R/W	Serial Flash Memory Operation Timing Adjustment Before timing adjustment: Set these bits to H’80000000. During timing adjustment: Set these bits to H’80000032. *Note*: Do not set any value other than the specified ones.

25.4.27 PHY Adjustment Register 2 (PHYADJ2)

PHYADJ2 is a 32-bit register used to adjust the timing when serial flash memory is connected.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADJ2[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADJ2[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

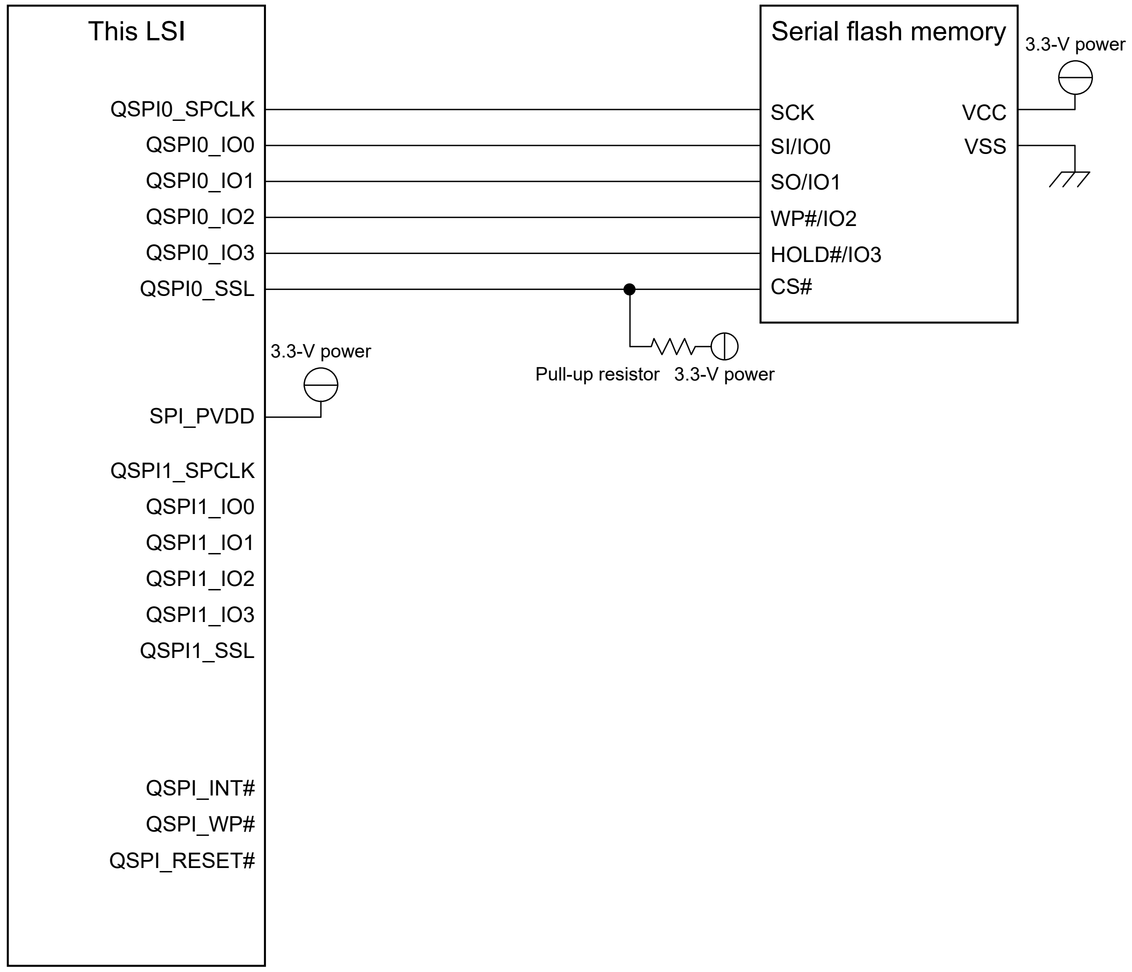
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADJ2[31:0]	All 0	R/W	Serial Flash Memory Operation Timing Adjustment Before timing adjustment: Set these bits to H'A5390000. During timing adjustment: Set these bits by following the procedure shown in **Figure 25.28, (1) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode)** and **Figure 25.29, (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode)**. *Note*: Do not set any value other than the specified ones.

25.5 Operation

25.5.1 System Configuration

This module can connect one or two serial flash memory devices, an Octal-SPI flash memory device, or a HyperFlash memory device. The number of connected memories can be selected using the BSZ[1:0] bits in CMNCR.

Examples of system configuration are shown in **Figure 25.2**, **Figure 25.3**, **Figure 25.4**, and **Figure 25.5** respectively.



Note: When connecting this LSI to a single serial flash memory device, use the pins for channel 0. The pins for channel 1 cannot be used in this configuration.

Figure 25.2 System Configuration Example with 4-Bit Data Size and One Serial Flash Memory Connected

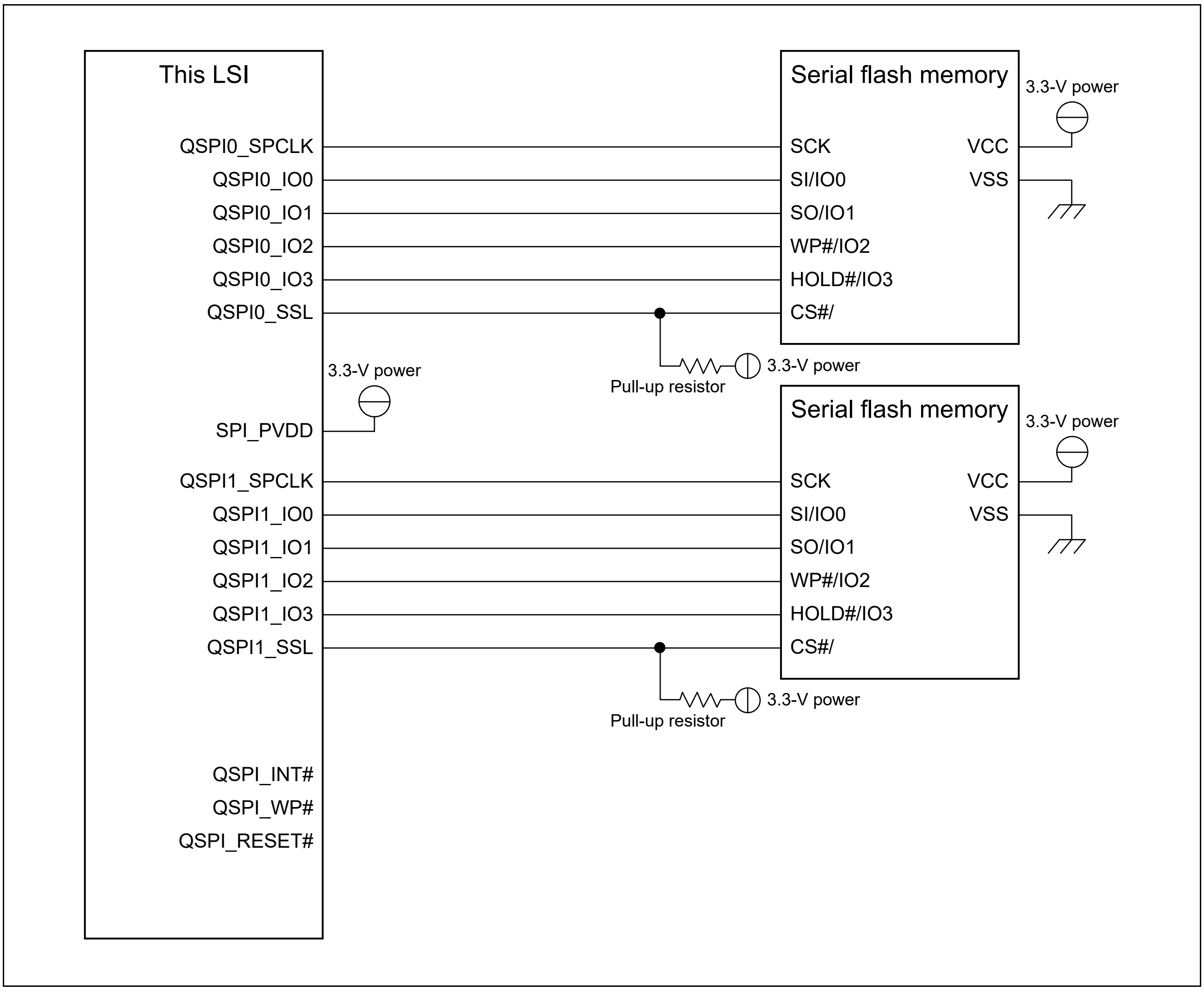


Figure 25.3 System Configuration Example with 4-Bit Data Size and Two Serial Flash Memories Connected (BSZ[1:0] Bits in CMNCR = 01)

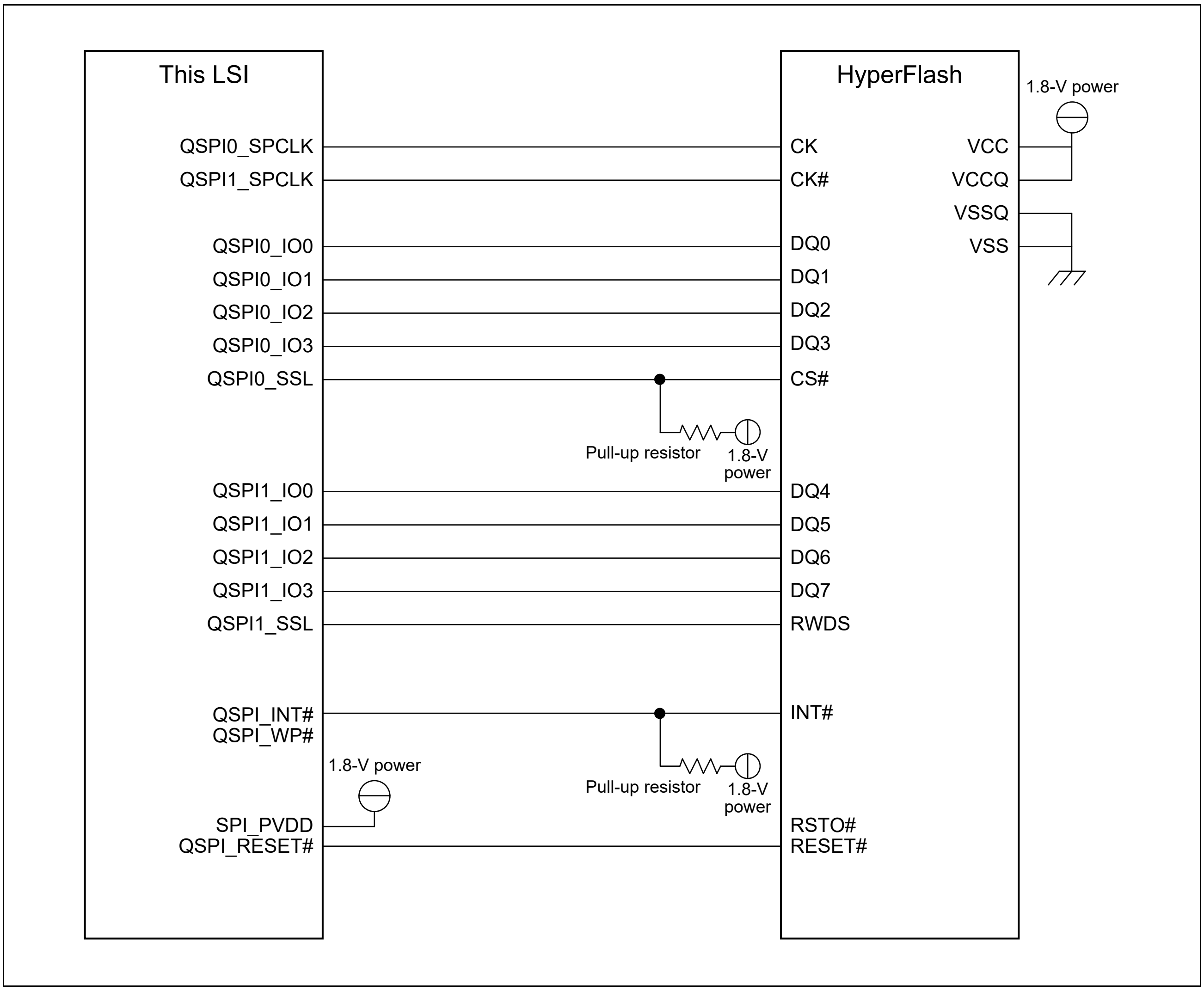


Figure 25.4 System Configuration Example with HyperFlash Connected

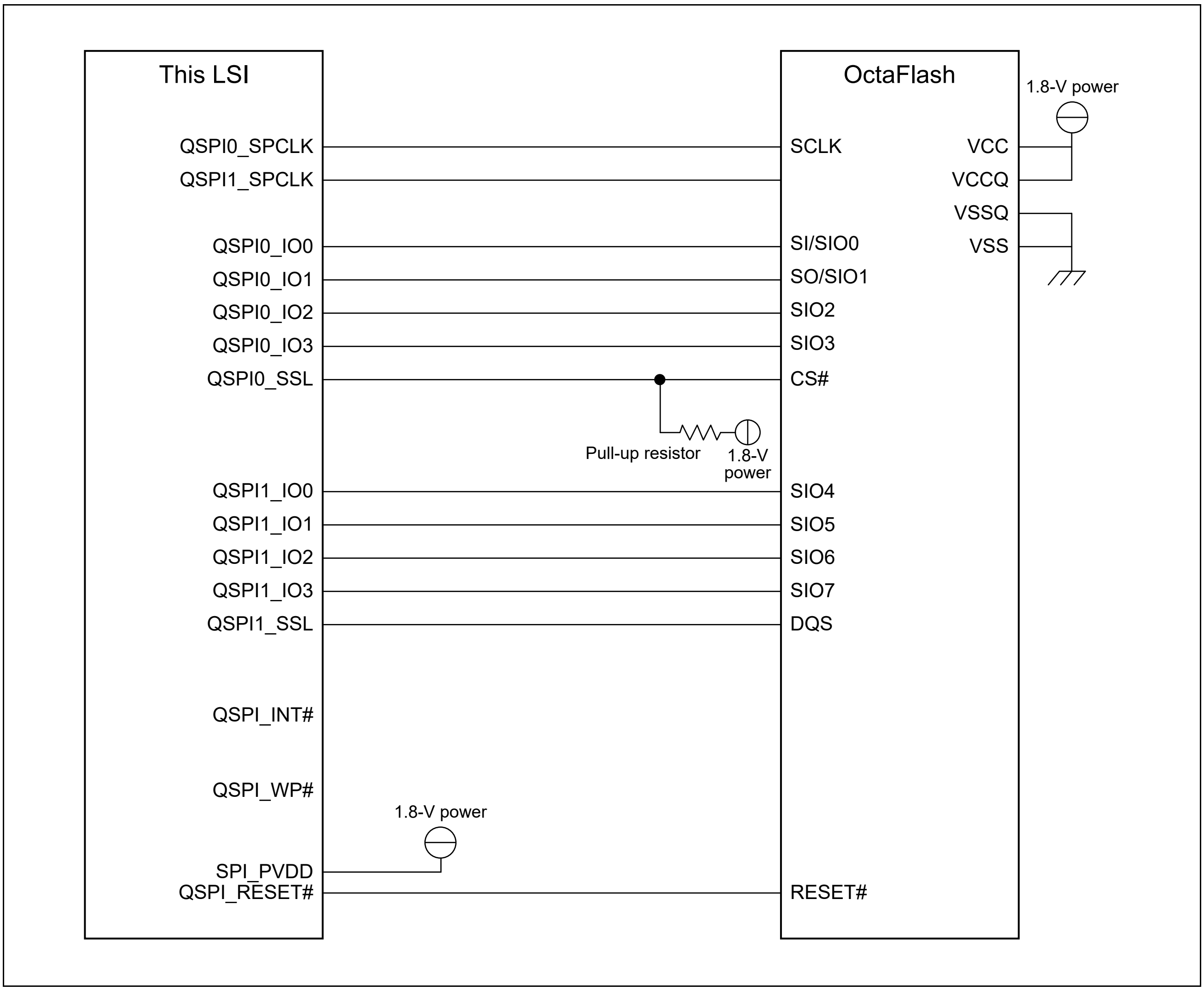


Figure 25.5 System Configuration Example with OctaFlash Connected

25.5.2 Address Map

In external address space read mode, the serial flash or HyperFlash connected is assigned in the SPI multi I/O bus space. In combination with DREAR, a maximum of 4 Gbytes can be accessed when one serial flash memory is connected, and a maximum of 8 Gbytes can be accessed when two serial flash memory devices or HyperFlash or Octal-SPI flash memory are connected.

Table 25.3 Address Map

Internal Address for Cortex-A55	Max. Access Area
H'0_20000000 to H'0_2FFFFFFF	One serial flash memory connected: 4 G-byte
	Two serial flash memory, HyperFlash or Octal-SPI flash memory connected: 8 G-byte

Internal Address for Cortex-M33 (Secure)	Max. Access Area
H'90000000 to H'9FFFFFFF	One serial flash memory connected: 4 G-byte
	Two serial flash memory, HyperFlash or Octal-SPI flash memory connected: 8 G-byte

Internal Address for Cortex-M33 (Non-Secure)	Max. Access Area
H'80000000 to H'8FFFFFFF	One serial flash memory connected: 4 G-byte
	Two serial flash memory, HyperFlash or Octal-SPI flash memory connected: 8 G-byte

Note: When access protection is set, not only this address space, but also the register area is protected.

25.5.3 32-bit Serial Flash Addresses

Since the SPI multi I/O bus space is 256 Mbytes, only a part of the 32-bit serial flash address area can be directly accessed. Here, the fixed value set in the pertinent register is used as the upper bit value of a 32-bit address.

To output serial flash addresses in 32 bits, set the ADE[3] bit in DRENR to 1, set the range of the external addresses used as the serial flash addresses to the EAC[2:0] bits in DREAR, and set the upper bit value of the 32-bit address as the fixed value to the EAV[7:0] bits in DREAR.

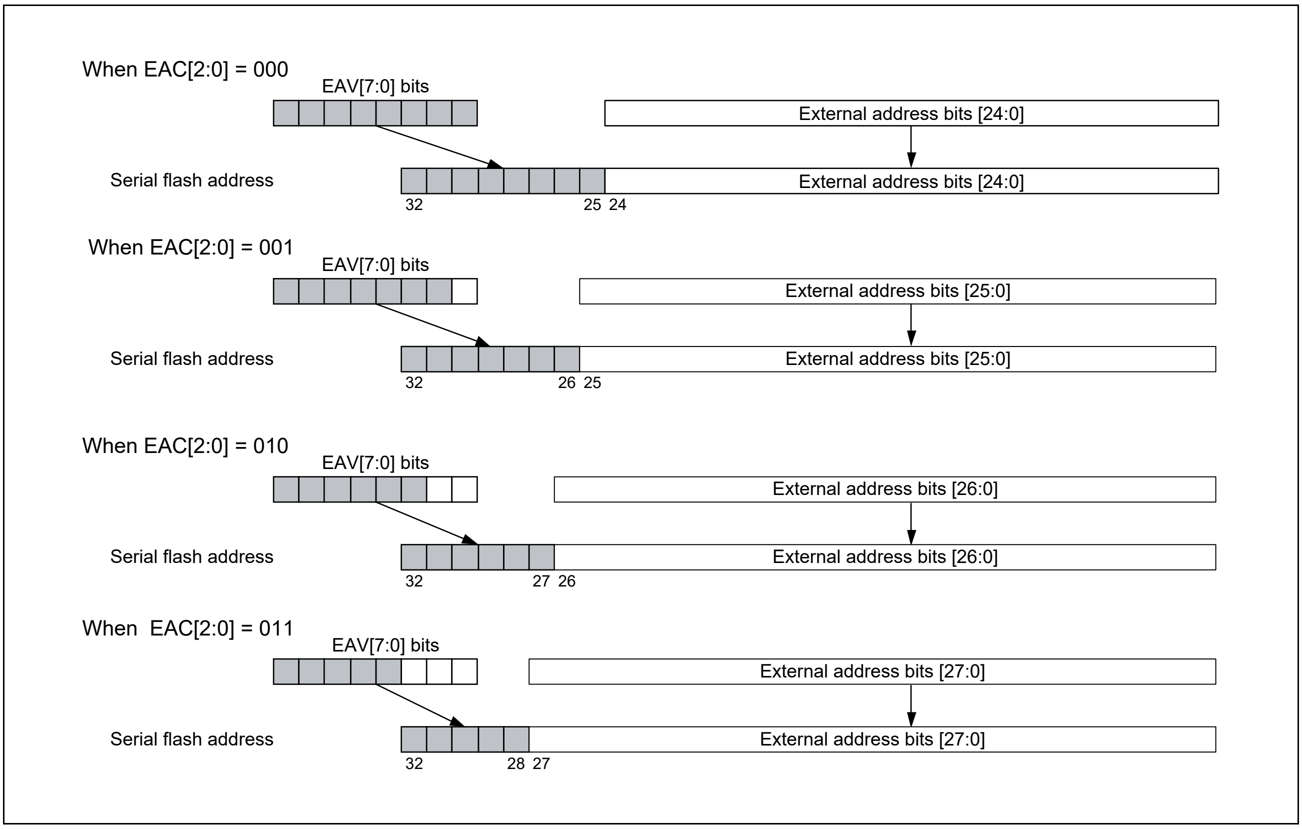


Figure 25.6 32-Bit Address Setting

Setting the ADE[3] bit in DRENR to 1 allows the serial flash address to be output using [31:0] bits.

When EAC[2:0] = 000, external address bits [24:0] are valid; set the value for [32:25] bits to EAV[7:0].
When EAC[2:0] = 001, external address bits [25:0] are valid; set the value for [32:26] bits to EAV[7:1].
When EAC[2:0] = 010, external address bits [26:0] are valid; set the value for [32:27] bits to EAV[7:2].
When EAC[2:0] = 011, external address bits [27:0] are valid; set the value for [32:28] bits to EAV[7:3].

The addresses actually used for external devices are as follows.

- Use the address [31: 0] when connecting one serial flash.
- Use address [32: 1] for two serial flash and Octal-SPI flash memory connections.
- Data read extended address setting is not available when connecting to HyperFlash. The area that can be directly accessed when connecting to HyperFlash is [25: 0] (64 Mbyte space). If you want to access outside the 64MB space, use manual mode.

25.5.4 Operating Modes

This module has two operating modes: external address space read mode and Manual mode.

In external address space read mode, a read access to the SPI multi I/O bus space is converted into Data read protocol and data is received. After data acquisition, data is returned to the bus master that is the issuing source. For details, see **Section 25.5.5, External Address Space Read Mode**.

In Manual mode, arbitrary protocol is carried out using register settings. For details, see **Section 25.5.7, Manual mode**.

25.5.5 External Address Space Read Mode

A read access to the SPI multi I/O bus space can be converted into Data read protocol in external address space read mode. Further, the commands, optional commands, option data, and dummy cycle issued for reading can be modified using registers.

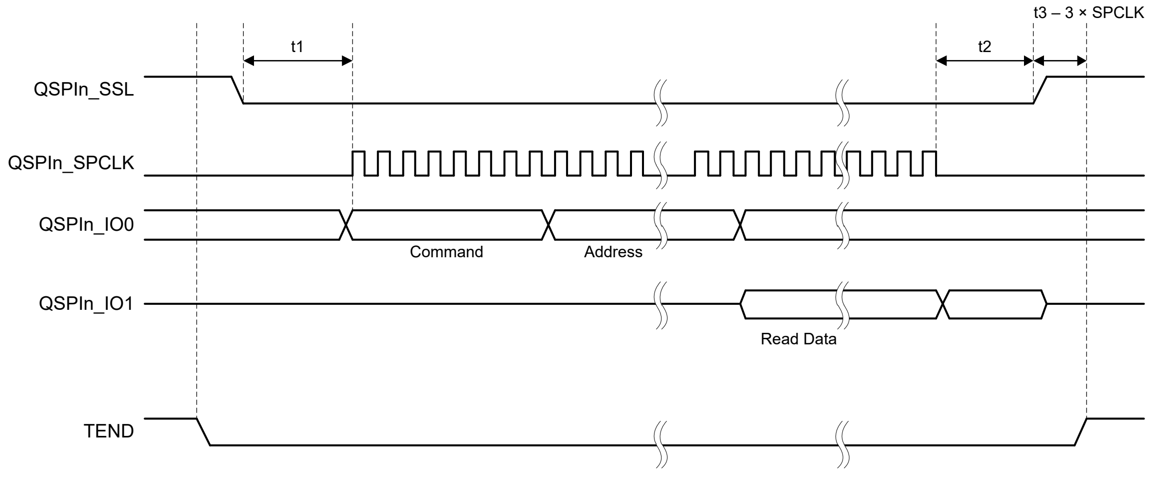
In external address space read mode, either normal read operation or burst read operation can be selected. The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), data read control register (DRCR), data read command setting register (DRCMR), data read extended address setting register (DREAR), data read option setting register (DROPR), data read enable setting register (DRENR), data read dummy cycle setting register (DRDMCR), and data read DDR enable register (DRDRENR).

(1) Normal Read Operation

When the RBE bit in the DRCR register is set to 0, normal read operation is performed. In the normal read operation, data is read from the external device when the bus master executes read access. After data is read, the QSPIn_SSL pins are negated.

The timing of normal read operation for the serial flash memory is shown in **Figure 25.7**.

t1 is the time period from QSPIn_SSL pin assertion to QSPIn_SPCLK oscillation (clock delay), t2 is the time period from transmission of the last QSPIn_CLK edge of a transfer to QSPIn_SSL pin negation (QSPIn_SSL negation delay), and t3 is the time period from one transfer end to the next transfer start (next access).



Note: t1, t2, and t3 are specified with SSLDR.SCKDL, SSLDR.SLNDL and SSLDR.SPNDL bits, respectively.

Figure 25.7 Normal Read Operation Timing

(2) Burst Read Operation

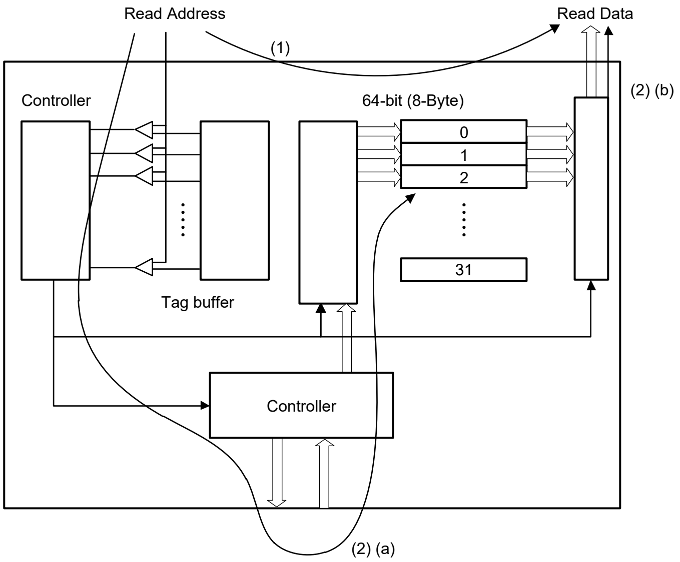
When the RBE bit in DRCR is set to 1, burst read operation is performed.

Read cache is enabled in the burst read operation. For read cache operation, see **Section 25.5.6, Read Cache**.

When the bus master executes read access, the read cache is first searched for the target data. When the read cache contains the data, it is read from the cache instead of accessing the external device. When the read cache does not contain the data, the external device is read in the burst mode and the read data is stored in the read cache and returned to the bus master. In this operation, the data transfer length is 64 bits × the value in the RBURST[4:0] bits, and data is always read from a 64-bit boundary address.

The QSPIn_SSL pin status after data transfer can be selected by using the SSLE bit in DRCR. When the SSLE bit is set to 0, the QSPIn_SSL pin is negated after data transfer. For an operation performed when the SSLE bit is set to 1, see **Section 25.5.5(3), Burst Read Operation with Automatic QSPIn_SSL Negation**, just below.

A pattern diagram of this operation and a burst read operation timing diagram when SSLE bit is set to 0 are shown in **Figure 25.8** and **Figure 25.9**.



(1) When the read cache contains the data
The data is read from the read cache without accessing external devices.

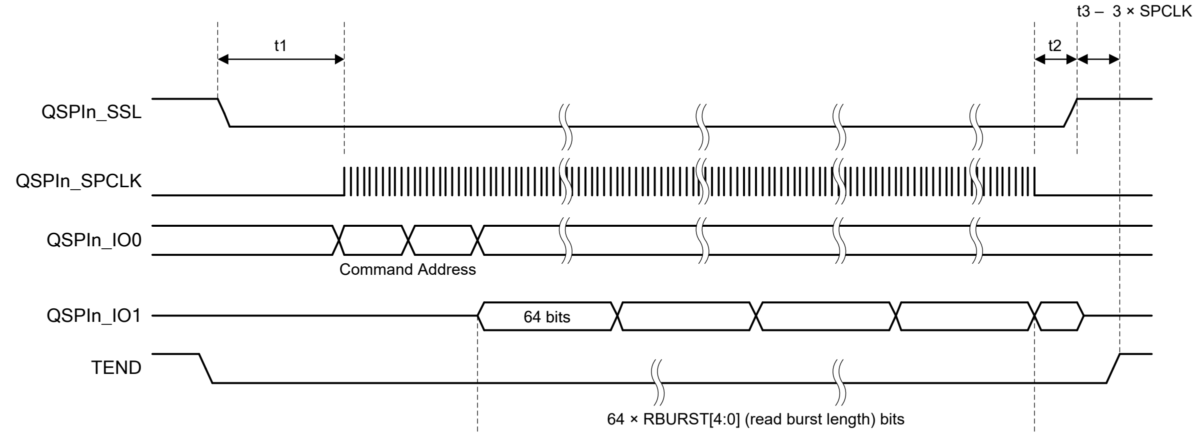
(2) When the read cache does not contain the data
(a) The data is read from the external device and stored in the read cache.
(b) The data is read from the read cache and returned to the bus master.

Figure 25.8 Burst Read Operation

NOTE

When connecting this LSI to a single serial flash memory device, use the pins for channel 0.

The pins for channel 1 cannot be used in this configuration.



Note: t1, t2, and t3 are specified with SSLDR.SCKDL, SSLDR.SLNDL and SSLDR.SPNDL bits, respectively.

Figure 25.9 Burst Read Operation Timing

(3) Burst Read Operation with Automatic QSPIn_SSL Negation

When SSLE bit in DRCR is set to 1, this module does not negate the QSPIn_SSL pin after the burst read transfer. When accessing the next time, if the address is continuous with the previous read address, the burst read operation is performed without issuing the command, optional command, address, option data, or dummy cycle. If the address is not continuous with the previous read address, the QSPIn_SSL pin is once negated and the burst read operation is performed after issuing the command, optional command, address, option data, or dummy cycle.

Burst read timing diagrams for continuous address and non-continuous address are shown in **Figure 25.10** and **Figure 25.11**.

For the next access after negation of the QSPIn_SSL with the SSLN bit in DRCR with this operation, read SSLF = 0 in CMNSR to confirm that the QSPIn_SSL has been negated.

This function is available when serial flash memory is connected.

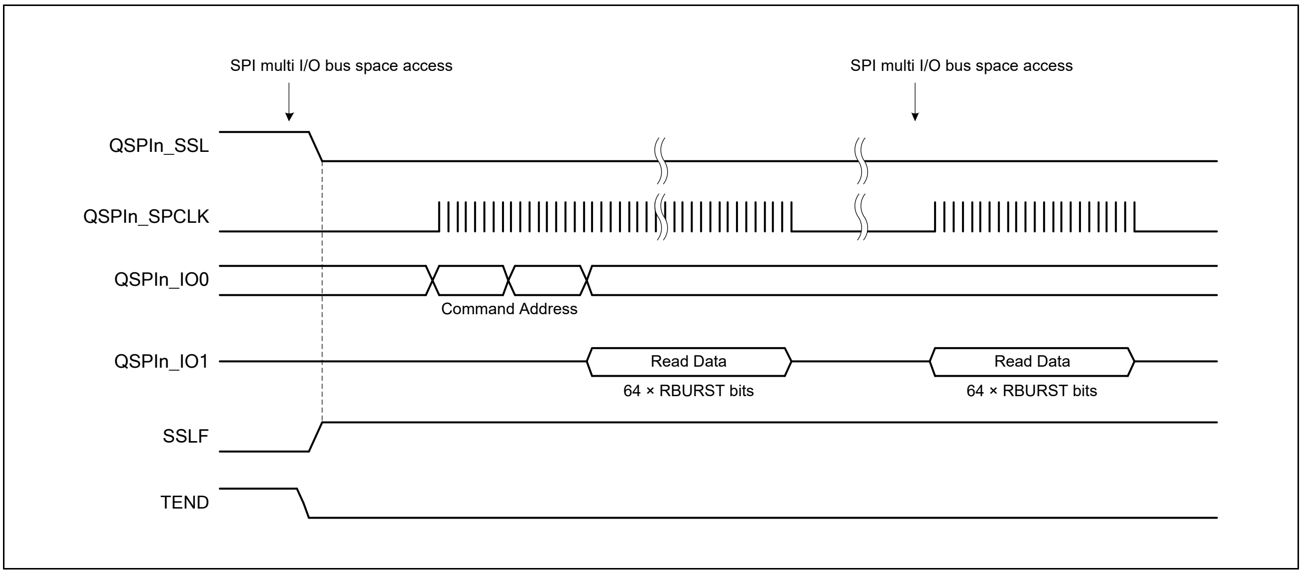


Figure 25.10 Burst Read Timing for Continuous Address (SSLE Bit = 1)

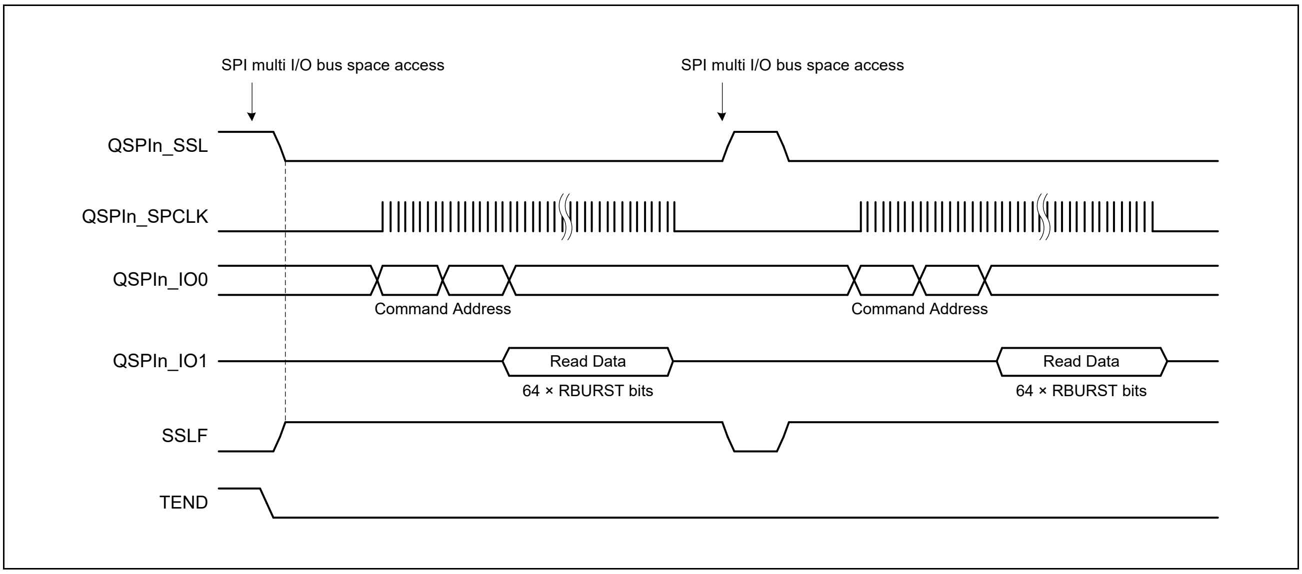
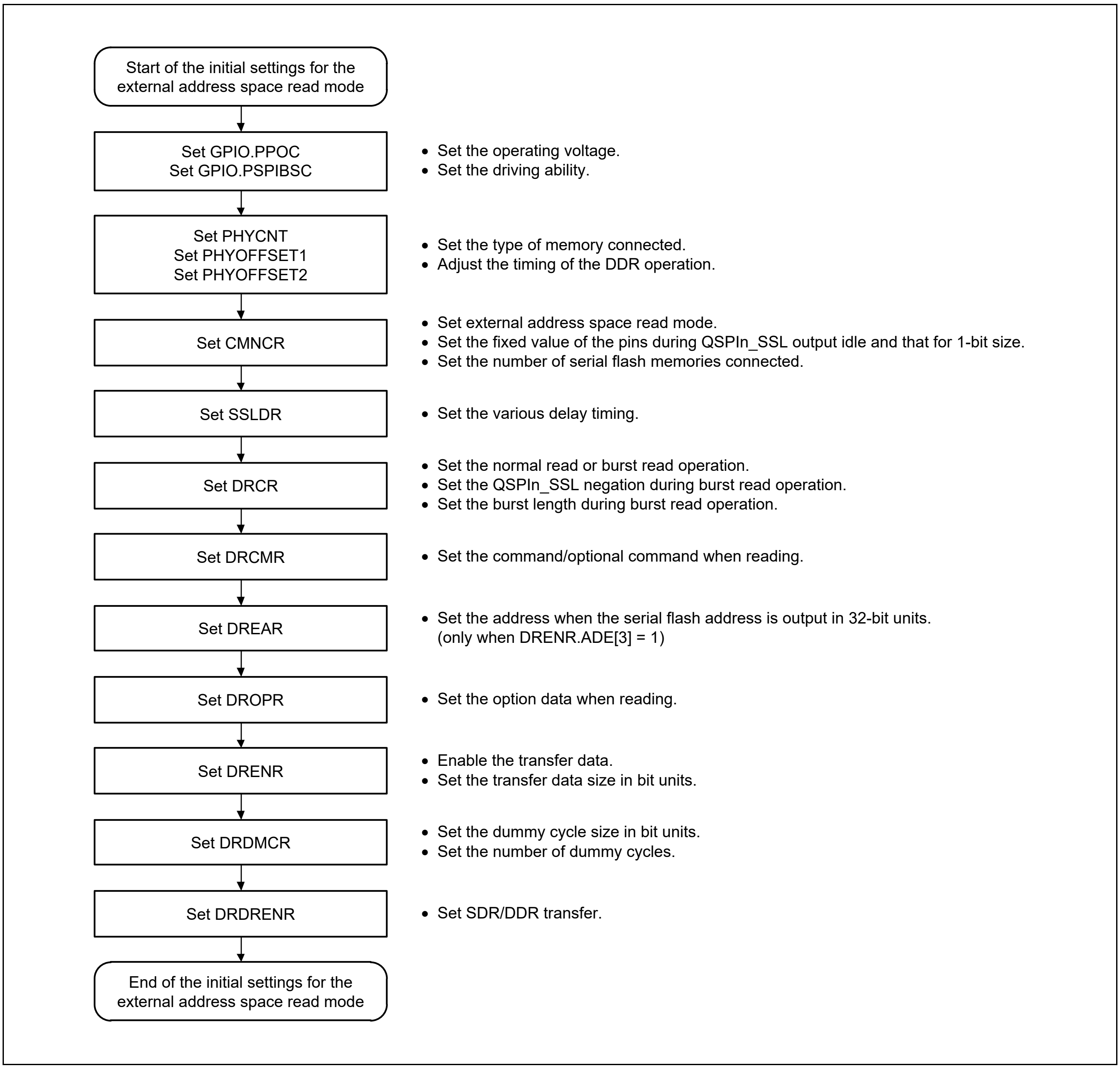


Figure 25.11 Burst Read Timing for Non-Continuous Address (SSLE Bit = 1)

(4) Initial Setting Flow

An example of an initial setting flow in external address space read mode is shown in **Figure 25.12**.



- Set the operating voltage.
- Set the driving ability.

- Set the type of memory connected.
- Adjust the timing of the DDR operation.

- Set external address space read mode.
- Set the fixed value of the pins during QSPIn_SSL output idle and that for 1-bit size.
- Set the number of serial flash memories connected.

- Set the various delay timing.

- Set the normal read or burst read operation.
- Set the QSPIn_SSL negation during burst read operation.
- Set the burst length during burst read operation.

- Set the command/optional command when reading.

- Set the address when the serial flash address is output in 32-bit units. (only when DRENR.ADE[3] = 1)

- Set the option data when reading.

- Enable the transfer data.
- Set the transfer data size in bit units.

- Set the dummy cycle size in bit units.
- Set the number of dummy cycles.

- Set SDR/DDR transfer.

Figure 25.12 Example of Initial Setting Flow in External Address Space Read Mode

25.5.6 Read Cache

This module has a simple built-in read cache. The read cache can be used during external address space read mode and burst read operation. The read cache is configured with a line size of 64 bits and 32 entries.

Read cache configuration is shown in **Figure 25.13**.

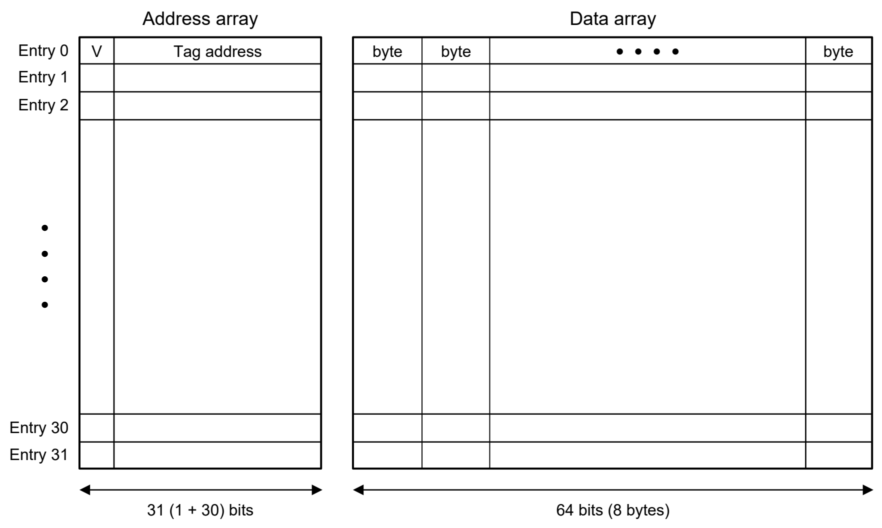


Figure 25.13 Read Cache Configuration

(1) Address Array

The V bit indicates whether the entry data is valid. When the V bit is 1, the data is valid and when V bit is 0, the data is invalid.

The tag address bits hold the address used for the serial flash memory. Address bits 32 to 3 are used for the purpose. Address bits 23 to 3 are enabled when address output is 24 bits and one serial flash memory is connected; and address bits 24 to 3 are enabled when two serial flash memories are connected.

Address bits 31 to 3 are enabled when address output is 32 bits and one serial flash memory is connected; and address bits 32 to 3 are enabled when two serial flash memories are connected.

(2) Data Array

It retains the 64-bit read data. Registration in the read cache is performed in line units.

(3) Read Operation

In case of read-hit, data is read from the read cache. In case of read-miss, after the 64 × RBURST (read burst length) data is read from the serial flash memory and the read cache is updated, the data is returned to the bus master.

(4) Data Replacement

The write pointer is used to update data. In case of read-miss, the RBURST (read burst length) portion data is replaced starting at the entry specified by the write pointer. In other words, the data is replaced in the storage order of the data. Whether data is referred to or not will not affect the replacement order of data.

25.5.7 Manual mode

This module can carry out an arbitrary serial transmission operation by using the register settings.

The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), Manual mode control register (SMCR), Manual mode command setting register (SMCMR), Manual mode address setting register (SMADR), Manual mode option setting register (SMOPR), and Manual mode enable setting register (SMENR), Manual mode read data register (SMRDR), Manual mode write data register (SMWDR), Manual mode dummy cycle setting register (SMDMCR), and Manual mode DDR enable register (SMDRENR).

Manual mode can be used for reading the status of the serial flash memory and writing to the serial flash memory.

In this mode, one transfer refers to the operation from when the SPIE bit in SMCR is set to 1 to when the TEND bit is set to 1.

(1) Transfer Start

The transfer of data is started in the set transfer format by setting the SPIE bit in SMCR to 1. When write operation is enabled, the Manual mode write data register is transmitted to the serial flash memory. When read operation is enabled, data read from the serial flash memory is stored into the Manual mode read data register.

The Manual operation timing is shown in **Figure 25.14**.

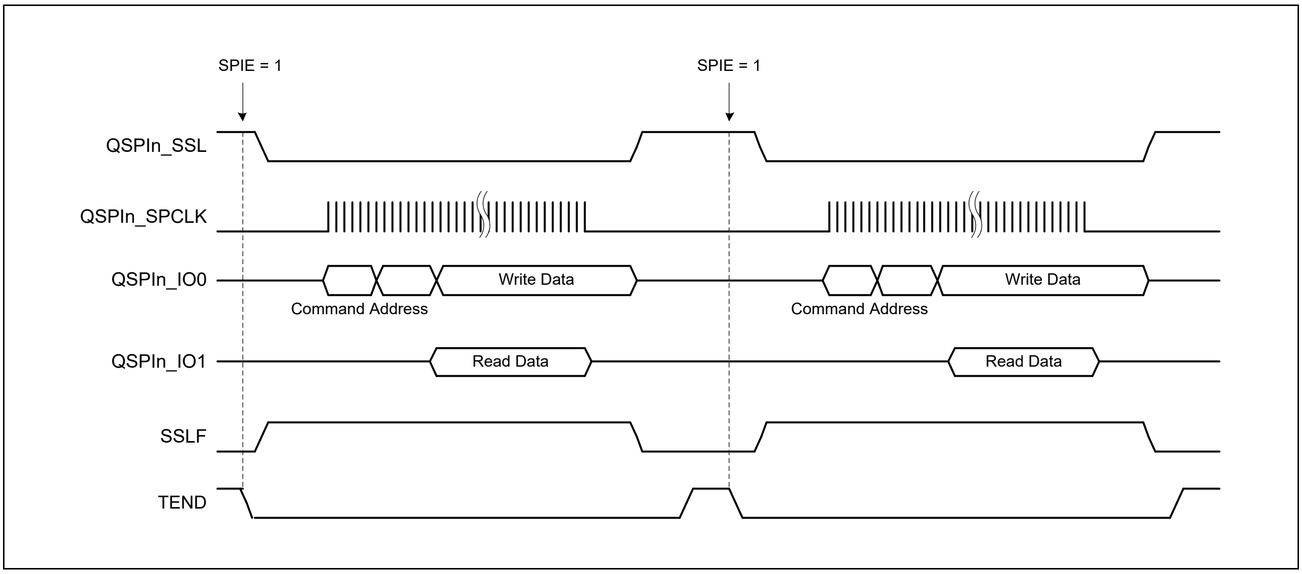


Figure 25.14 Serial transmission operation timing

(2) Read/Write Enable

- Read operation: Data can be read by setting the SPIRE bit in SMCR to 1. The read data is stored into SMRDR.
- Write operation: Data can be written by setting the SPIWE bit in SMCR to 1. The data stored in SMWDR is output.

When the data size is set to 1 bit using the SPIDB[1:0] bits in SMENR, data can be transmitted and received by setting the SPIRE and SPIWE bits to 1. However, when the data size is set to 4 bits by using the SPIDB[1:0] bits, only one of the SPIRE and SPIWE bits should be enabled. The operation is not guaranteed if the both bits are enabled.

(3) Retention of QSPIn_SSL Pin Assertion

By setting the SSLKP bit in SMCR to 1, assertion of the QSPIn_SSL pin can be continued till the next transfer. With this function, the transfer can be carried out continuously with the QSPIn_SSL kept in the asserted state.

The data transfer timing using the SSLKP bit is shown in **Figure 25.15**.

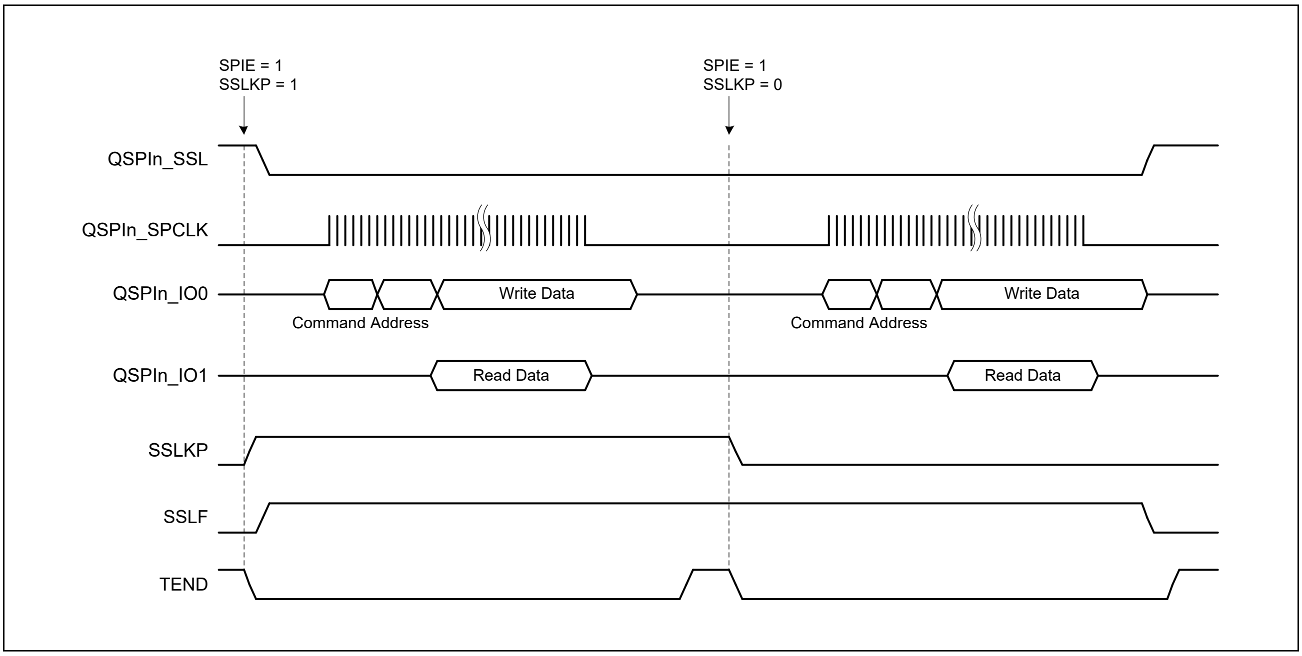


Figure 25.15 Data Transfer Timing using the SSLKP Bit

(4) Initial Setting Flow

An example of an initial setting flow in Manual mode is shown in **Figure 25.16**.

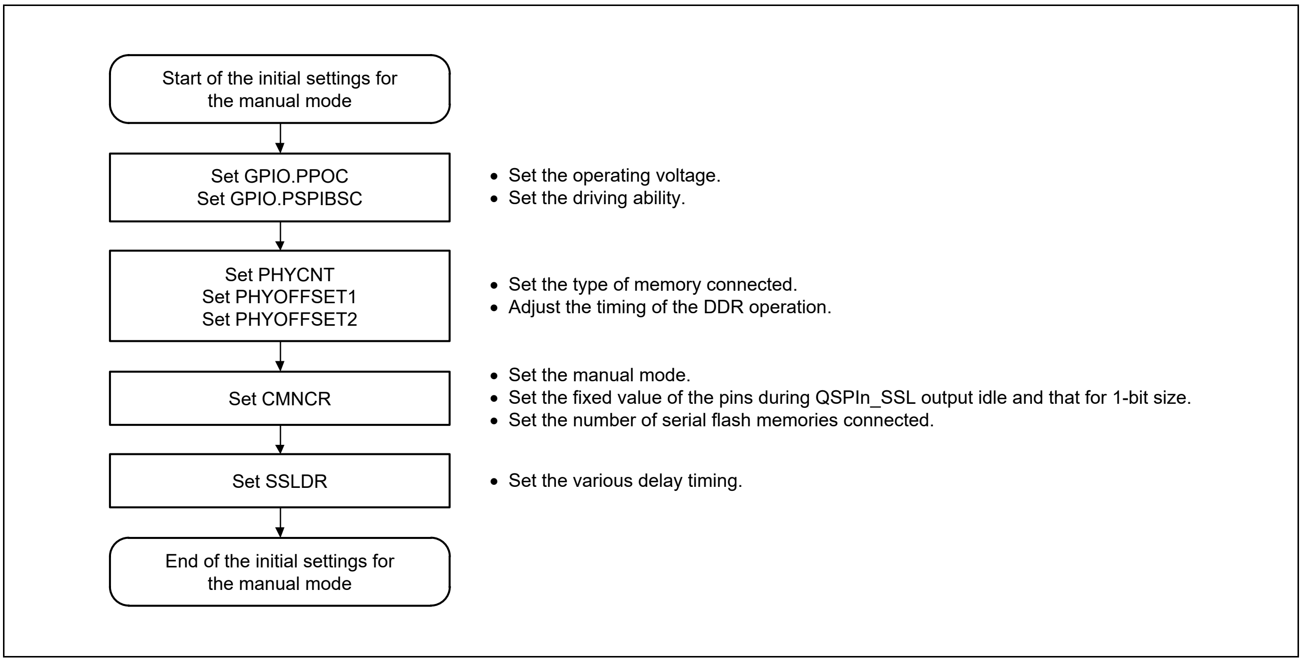
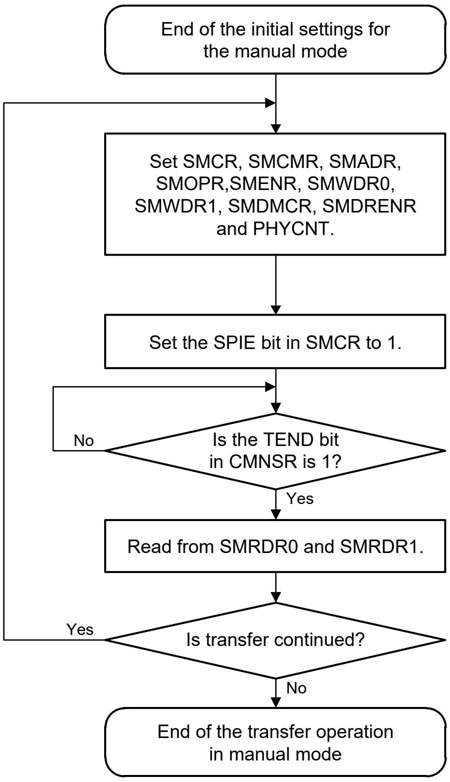


Figure 25.16 Example of Initial Setting Flow in Manual mode

(5) Data Transfer Setting Flow

An example of a data transfer setting flow in manual mode is shown in **Figure 25.17**.



- Set the QSPIn_SSL signal level to be kept.
- Enable data reading or data writing.
- Set the command/optional command/address/option data when reading.
- Enable the transfer data and set the transfer data size in bit units.
- Set the write data (valid when the SPIWE bit in SMCR is set to 1).
- Set the size and number of dummy cycles.
- Set SDR/DDR transfer.
- Specify whether to perform PHY calibration.
- Set the type of connected memory.

- Transfer the data when the SPIE bit is set to 1.

- The TEND bit is set to 1 on completion of the data transfer.

- Read the data (valid when the SPIRE bit in SMCR is set to 1).

Note: When data are transferred while the setting to retain the level of the QSPIn_SSL signal is made, release the setting at the time of transferring the last of the data.

Figure 25.17 Example of a Data Transfer Setting Flow in Manual Mode

25.5.8 Command Sequence

This module can input and output data in the order of command, optional command, address, option data, dummy cycle, and data.

(1) Data Registers

Table 25.4 shows the input and output data.

Table 25.4 Data Registers

Data		External Address Space Read Mode		Manual Mode
Command (8 bits)		CMD[7:0] bits in DRCMR		CMD[7:0] bits in SMCMR
Optional command (8 bits)		OCMD[7:0] bits in DRCMR		OCMD[7:0] bits in SMCMR
Address (32/24 bits)	BSZ[1:0] = 00 (one flash memory connected)	32 bits:	DREAR.EAV[6:3 to 0] bits + lower [27 to 24:0] bits of the read address.	32 bits: ADR[31:0] bits in SMADR 24 bits: ADR[23:0] bits in SMADR
		24 bits:	Lower [23:0] bits of the read address.	
	BSZ[1:0] = 01 (two flash memories connected)	32 bits:	DREAR.EAV[7:3 to 0] bits + lower [27 to 24:1] bits of the read address.	
		24 bits:	Lower [24:1] bits of the read address.	
Option data (8 bits × 4)		DROPR		SMOPR
Dummy cycle (2 to 20 cycles)		DRDMCR		SMDMCR (only when read)
Transfer data		Normal read: 8, 16, and 32 bits Burst read: 64 × RBURST bits		Read: SMRDR0, SMRDR1 Write: SMWDR0, SMWDR1

Table 25.5 Data Registers (HyperFlash)

Data	External Address Space Read Mode	Manual Mode
CA0 (47 to 40)	DRCMR.CMD[7:5] bits + [31:27] bits of the read address.	SMCMR.CMD[7:5] + SMADR.ADR[31:27]
CA0 (39 to 32)	[26:19] bits of the read address.	ADR[26:19] bits in SMADR
CA1 (31 to 24)	[18:11] bits of the read address.	ADR[18:11] bits in SMADR
CA1 (23 to 16)	[10:3] bits of the read address.	ADR[10:3] bits in SMADR
CA2 (15 to 8)	[15:8] bits in DROPR	[15:8] bits in SMOPR
CA2 (7 to 0)	DROPR [7:3] bits + [2:0] bits of the read address.	SMOPR [7:3] bits + SMADR.ADR[2:0] bits
Dummy cycle (2 to 20 cycles)	DRDMCR	SMDMCR
Transfer data	Normal read: 8, 16, and 32 bits Burst read: 64 × RBURST bits	Read: SMRDR0, SMRDR1 Write: SMWDR0, SMWDR1

(2) Data Enable

In external address space read mode, enabling or disabling transfer of the command, optional command, address, option data, and dummy cycle can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in DRENR, respectively. The number of dummy cycles can be controlled with the data read dummy cycle setting register (DRDMCR).

Similarly, in manual mode, enabling or disabling the command, optional command, address, option data, dummy cycle, and transfer data can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], DME, and SPIDE[3:0] bits in SMENR, respectively. However, disabling all the above parameters is prohibited in manual mode. At least one of them except dummy cycle must be enabled. The number of dummy cycles can be controlled with the Manual mode dummy cycle setting register (SMDMCR).

For the address and option data in external address space read mode and the address, option data, and transfer data in manual mode, the enable bit setting allowed is determined according to the transfer data size. For the allowed setting combinations of the enable bits and transfer data size, refer to the description of the pertinent register.

If data is disabled, that data is skipped, and input and output of the next data is carried out. The command, optional command, address, and option data are always output. During dummy cycles, the state of the used pins is Hi-Z. In external address space read mode, data is always input; and in manual mode, input and output of data is determined based on the settings of the SPIRE and SPIWE bits in SMCR.

There are some restrictions on dummy cycle insertion; refer to the description of the DME bits in DRENR and SMENR for details.

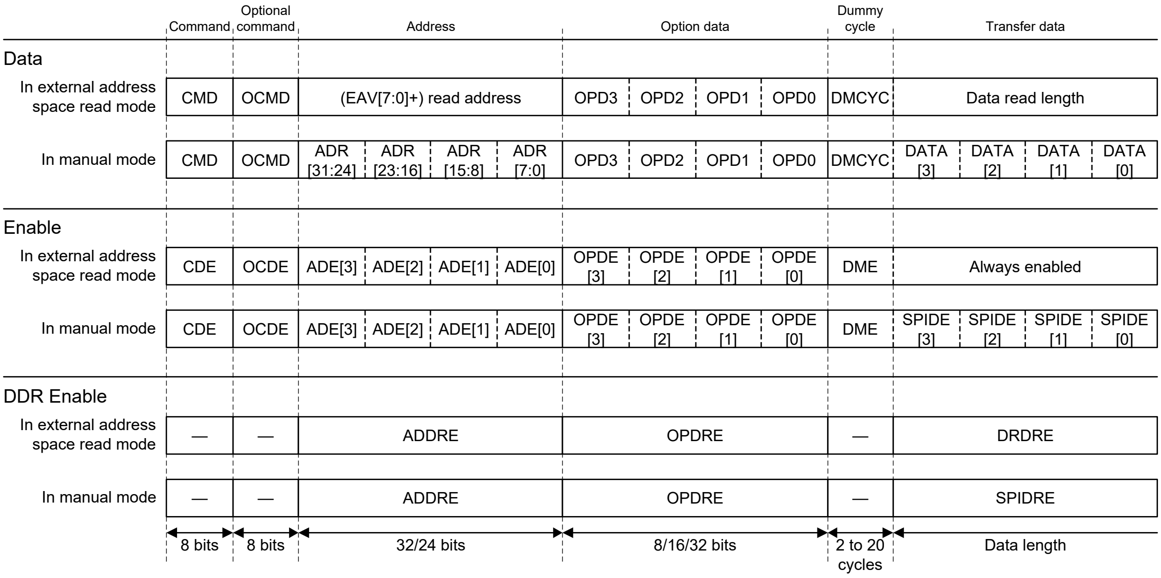


Figure 25.18 Data and Enable (Serial Flash)

When the HyperFlash memory is connected to this LSI, the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in the DRENR register and the HYPE[2:0], ADDRE, and OPDRE bits in the DRDRENR register should be set to “enabled” in external address space read mode. In Manual mode, the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in the SMENR register and the HYPE[2:0], ADDRE, OPDRE, and DRDRE bits in the SMDRENR should be set to “enabled”.

Table 25.6 Enable Register (HyperFlash)

External Address Space Read Mode			Manual Mode		
Register	Bit	Setting Value	Register	Bit	Setting Value
DRENR	CDE	1	SMENR	CDE	1
	OCDE	1		OCDE	1
	ADE[3:0]	0100		ADE[3:0]	0100
	OPDE[3:0]	0000		OPDE[3:0]	0000
	DME	1		DME	Read: 1 Write: 0
				SPIDE[3:0]	
				16 bits	1000
				32 bits	1100
				64 bits	1111
DRDRENR	HYPE[2:0]	101	SMDRENR	HYPE[2:0]	101
	ADDRE	1		ADDRE	1
	OPDRE	0		OPDRE	0
	DRDRE	1		SPIDRE	1

Table 25.7 Enable Register (Octal-SPI Flash Memory)

External Address Space Read Mode			Manual Mode		
Register	Bit	Setting Value	Register	Bit	Setting Value
DRENR	CDE	1	SMENR	CDE	1
	OCDE	0		OCDE	0
	ADE[3:0]	1100		ADE[3:0]	1100
	OPDE[3:0]	0000		OPDE[3:0]	0000
	DME	1		DME	Read: 1 Write: 0
SMENR	CDE	0		SPIDE[3:0]	
				16 bits	1000
				32 bits	1100
				64 bits	1111
DRDRENR	HYPE[2:0]	100, 101	SMDRENR	HYPE[2:0]	100, 101
	ADDRE	1		ADDRE	1
	OPDRE	0		OPDRE	0
	DRDRE	1		SPIDRE	1

Note: When only commands are to be transferred, set DRDRENR.HYPE or SMDRENR.HYPE to 100.

(3) Bit Size

In external address space read mode, the size of the command, optional command, address, option data, and the read data in bit units is respectively controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and DRDB[1:0] bits in DRENR.

Similarly, in manual mode, the size of the command, optional command, address, option data, and read write data in bit units is controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and SPIDB[1:0] bits in SMENR.

(a) 1-bit Size

When the size is set to 1 bit, QSPI0_IO1 and QSPI1_IO1 pins will be the input pins and QSPI0_IO0 and QSPI1_IO0 pins will be the output pins. QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 pins are not used.

Figure 25.19 and **Figure 25.20** show the transfer format examples.

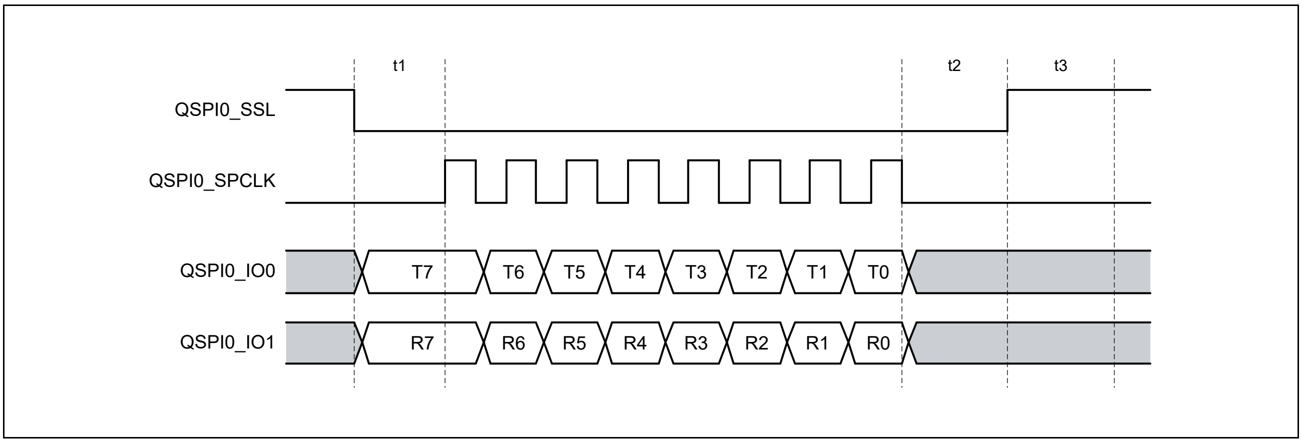


Figure 25.19 Transfer Format Example with 1-Bit Data Size and One Serial Flash Memory Connected

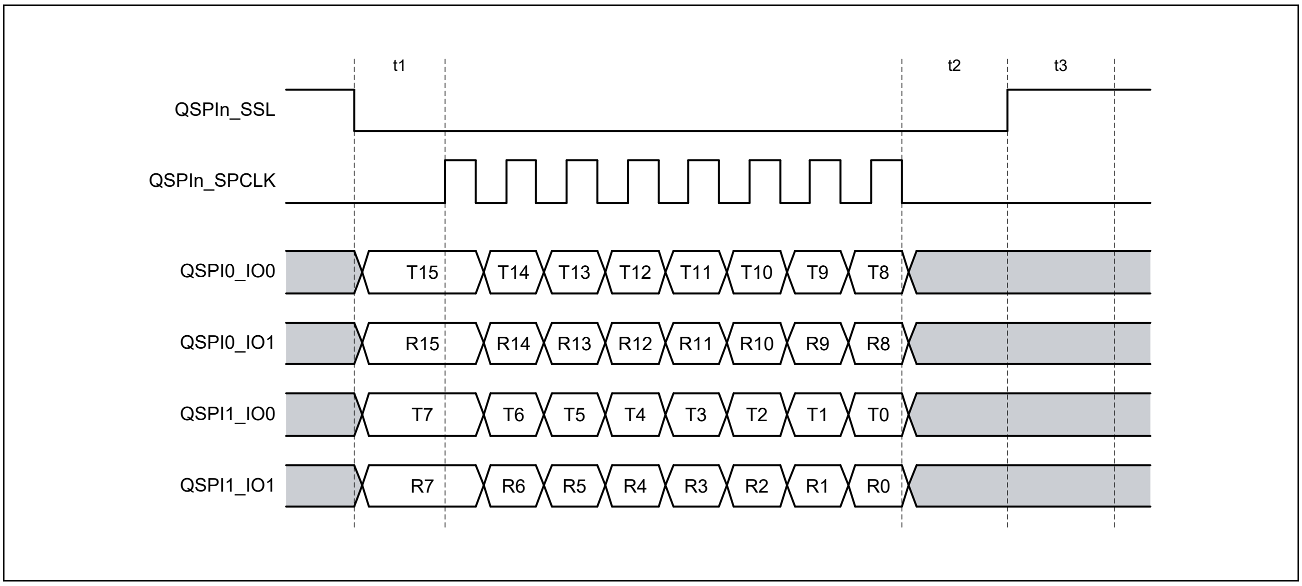


Figure 25.20 Transfer Format Example with 1-Bit Data Size and Two Serial Flash Memories Connected

(b) 4-bit Size (Serial Flash)

When the size is set to 4 bits, QSPI0_IO0, QSPI1_IO0, QSPI0_IO1, QSPI1_IO1, QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 pins will be either the input pins or the output pins.

Figure 25.21 and **Figure 25.22** show the transfer format examples.

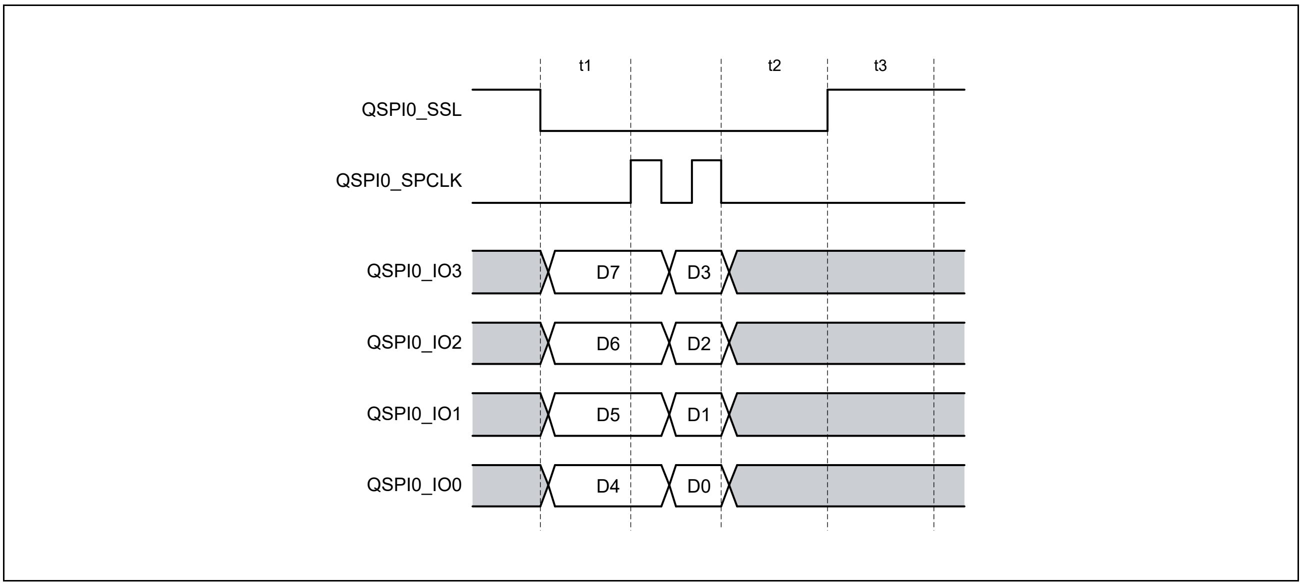


Figure 25.21 Transfer Format Example with 4-Bit Data Size and One Serial Flash Memory Connected

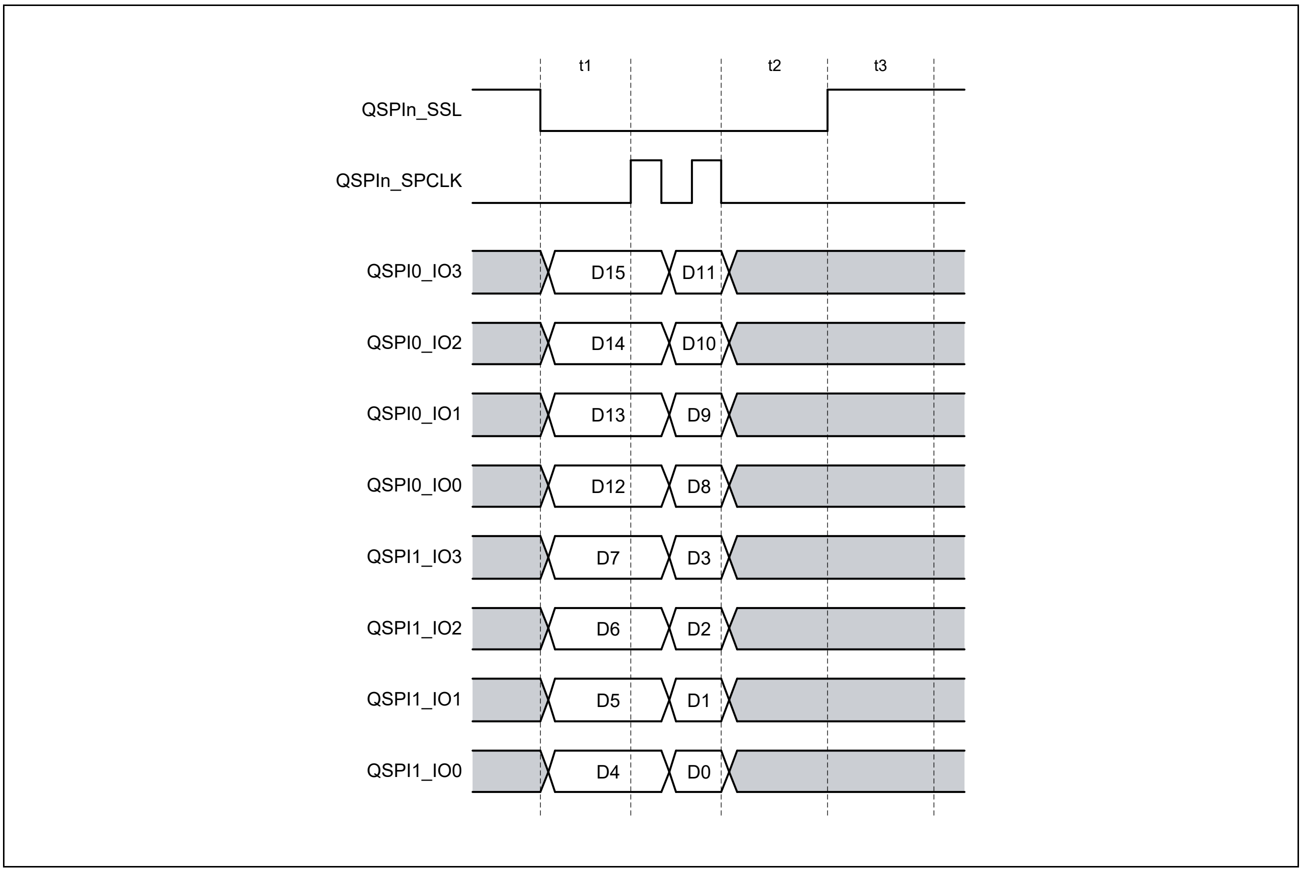


Figure 25.22 Transfer Format Example with 4-Bit Data Size and Two Serial Flash Memories Connected

(c) 8-bit Size (HyperFlash)

When the HyperFlash memory is connected to this LSI, QSPI0_SPCLK and QSPI1_SPCLK work as differential clock pins, QSPI1_SSL works as the read data strobe (RDS) input pin, and QSPI0_IO0, QSPI1_IO0, QSPI0_IO1, QSPI1_IO1, QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 work as either input pins or output pins.

Figure 25.23 and **Figure 25.24** show the transfer format examples.

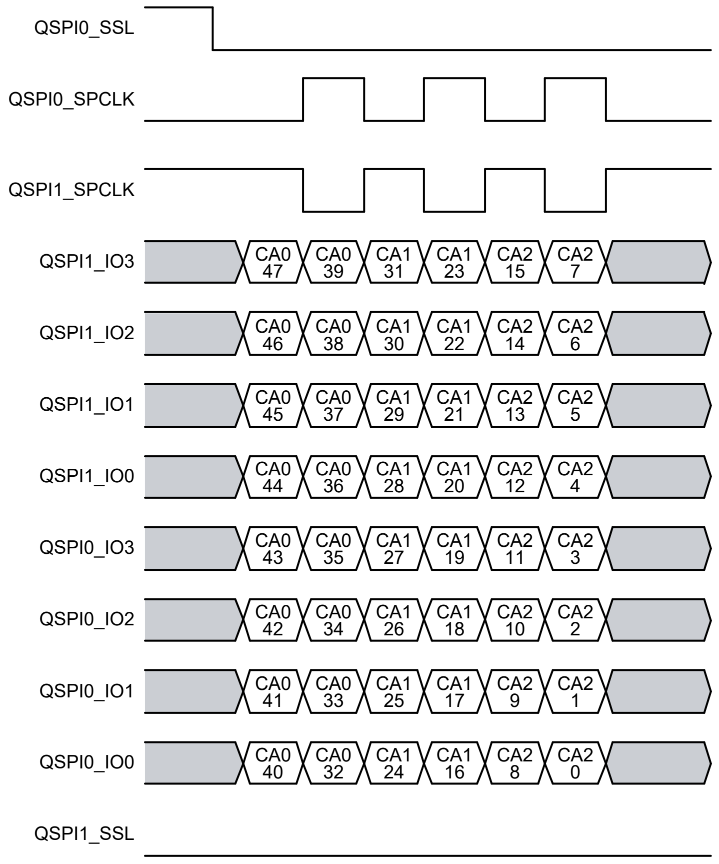


Figure 25.23 Transfer Format Example in the HyperFlash Command and Address Phase

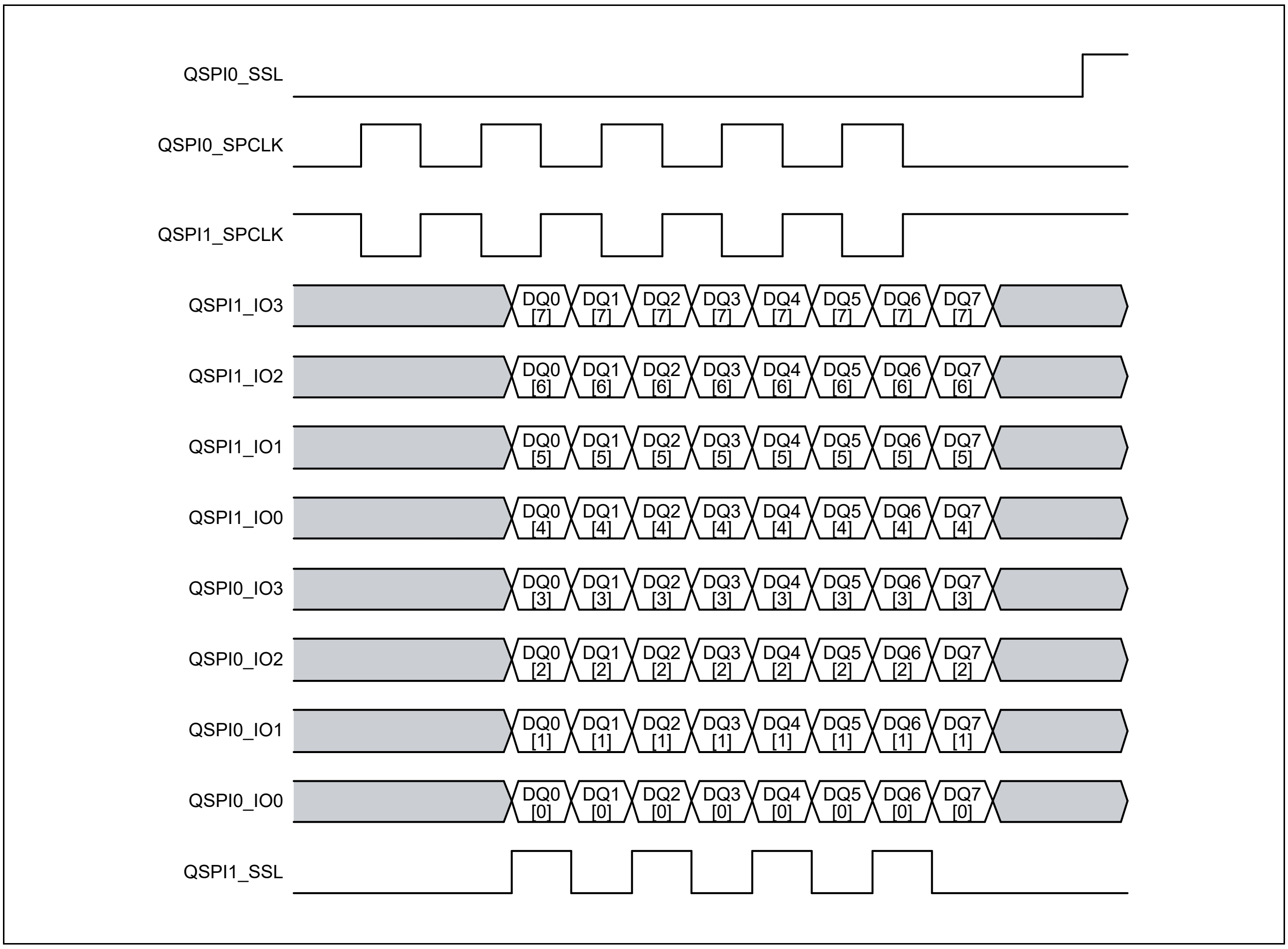


Figure 25.24 Transfer Format Example in the HyperFlash Read Phase

25.5.9 Data Pin Control

With this module, the status of pins can be automatically changed based on the data size to be used and the read/write settings. The pin status during the QSPIn_SSL negation can be set with the MOIIO3, MOIIO2, MOIIO1, and MOIIO0 bits in CMNCR. The QSPIn_SSL and QSPIn_SPCLK pins are always output pins.

The status of respective pins is specified in **Table 25.8** to **Table 25.11**.

When using only a single serial flash memory device, note that connecting the pins for channel 1 to the serial flash memory is prohibited.

Table 25.8 Pin Status (1)

		QSPIn_SSL Assertion	
		Command, Optional Command, Address, Option Data	
Pin	QSPIn_SSL Negation	1-bit Size	4-bit Size
QSPI0_IO0, QSPI1_IO0	MOIIO0 bit value	Output	Output
QSPI0_IO1, QSPI1_IO1	MOIIO1 bit value	Hi-Z	Output
QSPI0_IO2, QSPI1_IO2	MOIIO2 bit value	IO2FV bit value	Output
QSPI0_IO3, QSPI1_IO3	MOIIO3 bit value	IO3FV bit value	Output

Table 25.9 Pin Status (2)

	Transfer Data			
	External Address Space Read Mode		SPI Operation	
			SPIRE Bit = 1, SPIWE Bit = 0	
Pin	1-bit Size	4-bit Size	1-bit Size	4-bit Size
QSPI0_IO0, QSPI1_IO0	IO0FV bit value	Input	IO0FV bit value	Input
QSPI0_IO1, QSPI1_IO1	Input	Input	Input	Input
QSPI0_IO2, QSPI1_IO2	Hi-Z	Input	Hi-Z	Input
QSPI0_IO3, QSPI1_IO3	Hi-Z	Input	Hi-Z	Input

Table 25.10 Pin Status (3)

	Transfer Data			
	SPI Operation			
	SPIRE Bit = 0, SPIWE Bit = 1		SPIRE Bit = 1, SPIWE Bit = 1	
Pin	1-bit Size	4-bit Size	1-bit Size	4-bit Size
QSPI0_IO0, QSPI1_IO0	Output	Output	Output	Setting prohibited
QSPI0_IO1, QSPI1_IO1	Hi-Z	Output	Input	Setting prohibited
QSPI0_IO2, QSPI1_IO2	IO2FV bit value	Output	IO2FV bit value	Setting prohibited
QSPI0_IO3, QSPI1_IO3	IO3FV bit value	Output	IO3FV bit value	Setting prohibited

Table 25.11 Pin Status (4)

Pin	Dummy Cycle
QSPI0_IO0, QSPI1_IO0	IO0FV bit value
QSPI0_IO1, QSPI1_IO1	Hi-Z
QSPI0_IO2, QSPI1_IO2	Hi-Z
QSPI0_IO3, QSPI1_IO3	Hi-Z

25.5.10 QSPIn_SSL Pin Control

Negation conditions of the QSPIn_SSL pin are as follows.

(1) External Address Space Read Mode

(a) Normal read operation (RBE bit in DRCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(b) Burst read without automatic QSPIn_SSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(c) Burst read with automatic QSPIn_SSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 1)

- QSPIn_SSL negated after t2 cycle when the read address is not continuous with the previously read address
- QSPIn_SSL negated after the SSLN bit in DRCR is set to 1

(2) Manual Mode

(a) QSPIn_SSL pin assertion not retained (SSLKP bit in SMCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(b) QSPIn_SSL pin assertion retained (SSLKP bit in SMCR = 1)

QSPIn_SSL not negated.

When to be negated, data should be transferred after setting the SSLKP bit to 0.

NOTE

When HyperFlash or Octal-SPI flash memory is connected, the QSPIn_SSL pin is used as an input terminal for data strobe.

25.5.11 QSPIn_SPCLK Pin Control

QSPI0_SPCLK and QSPI1_SPCLK are output according to the following settings.

Table 25.12 QSPIn_SPCLK Output

Pin	HyperFlash connected. (CMNCR.BSZ = 01, PHYCNT.PHYMEM = 11)	One serial flash memory connected. (CMNCR.BSZ = 00, PHYCNT.PHYMEM = 00, 01)	Two serial flash memories connected. (CMNCR.BSZ = 01, PHYCNT.PHYMEM = 00, 01)
QSPI0_SPCLK	Output	Output	Output
QSPI1_SPCLK	Output (Complementary output of QSPI0_SPCLK)	Hi-Z	Output

25.5.12 Flags

This module has two flag bits SSLF and TEND in CMNSR. These bits are read-only bits.

(1) SSLF Bit

This bit indicates the QSPIn_SSL pin status. The status is 1 when the QSPIn_SSL is asserted, and the status is 0 when the QSPIn_SSL is negated.

(2) TEND Bit

This bit indicates whether transfer of data is in progress or the transfer of data has ended.

During t1 time period, data transfer, t2 time period, t3 time period, and waiting for read access by burst read and QSPIn_SSL automatic negation, the TEND bit is read as 0 to indicate that the transfer of data is in progress.

Otherwise, the TEND bit is read as 1 to indicate that transfer of data has ended.

(3) Register Re-writing Timing

The status of the TEND bit determines the rewritable registers.

The registers which can be written to, except the SSLN bit in DRCR, should be modified when TEND = 1. Read SMRDR0 and SMRDR1 when TEND = 1. CMNSR can always be read.

25.5.13 Write Buffer Operation

This module uses the read cache space also as the write buffer space during write operation. The write buffer improves the performance of write access.

Table 25.13 Write Buffer Space

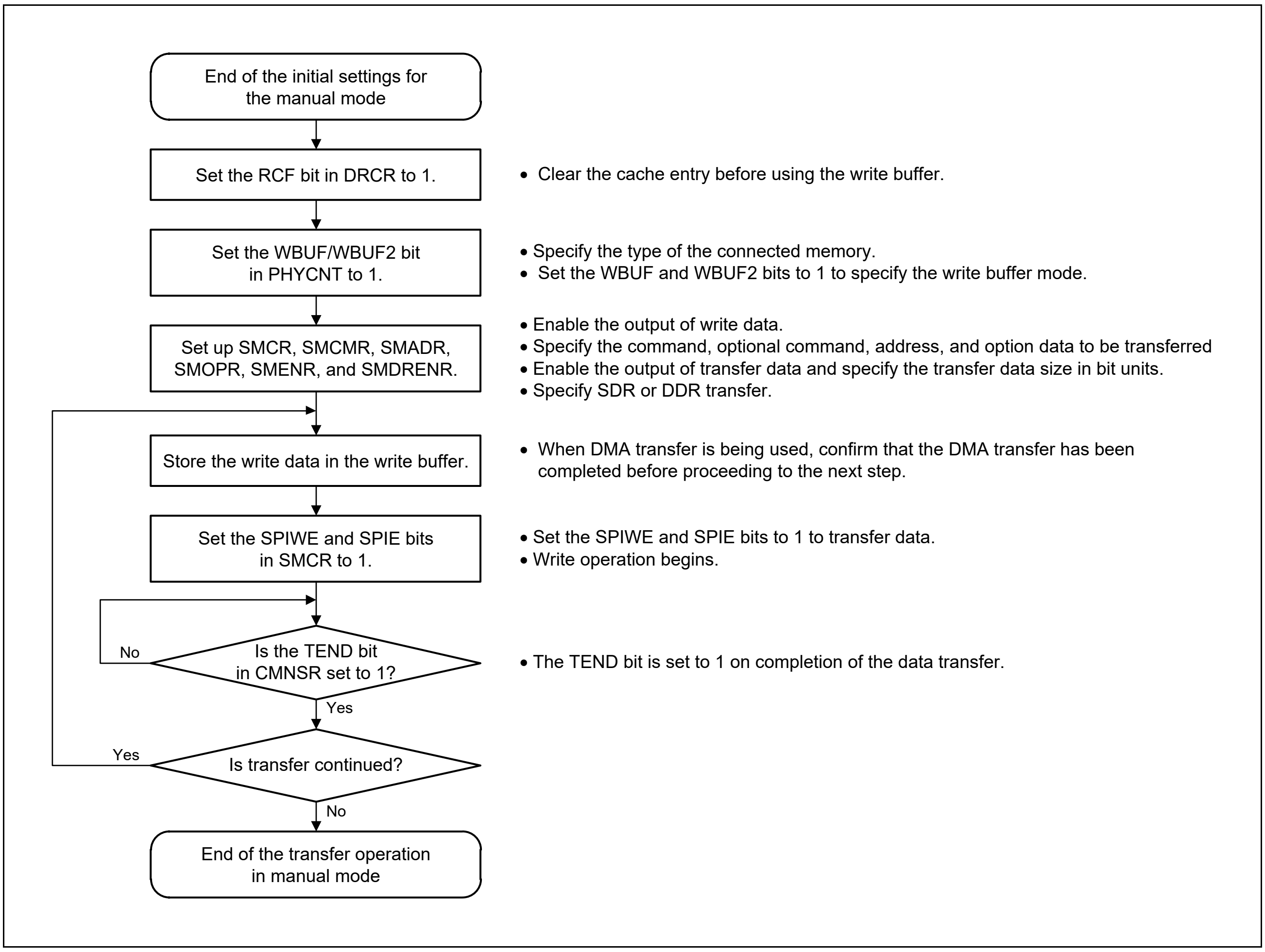
Address for Cortex-A55	Access Size
H'0_1007 0000 ~ H'0_1007 00FF	4, 8, 16, 32, or 64 bytes

Address for Cortex-M33 (Secure)	Access Size
H'5007 0000 ~ H'5007 00FF	4, 8, 16, 32, or 64 bytes

Address for Cortex-M33 (Non-Secure)	Access Size
H'4007 0000 ~ H'4007 00FF	4, 8, 16, 32, or 64 bytes

Note: Be sure to sequentially access this space beginning from its start address with the transfer size set to eight bytes.
If access is non-sequential or random, correct operation is not guaranteed.
After an operation in Manual mode or with the use of the cache area as the write buffer, clearing of this cache area by writing 1 to the RCF bit is required.

Figure 25.25 shows the procedure for using the write buffer. For the initial settings in Manual mode, see **Figure 25.16**.



- Clear the cache entry before using the write buffer.

- Specify the type of the connected memory.
- Set the WBUF and WBUF2 bits to 1 to specify the write buffer mode.

- Enable the output of write data.
- Specify the command, optional command, address, and option data to be transferred
- Enable the output of transfer data and specify the transfer data size in bit units.
- Specify SDR or DDR transfer.

- When DMA transfer is being used, confirm that the DMA transfer has been completed before proceeding to the next step.

- Set the SPIWE and SPIE bits to 1 to transfer data.
- Write operation begins.

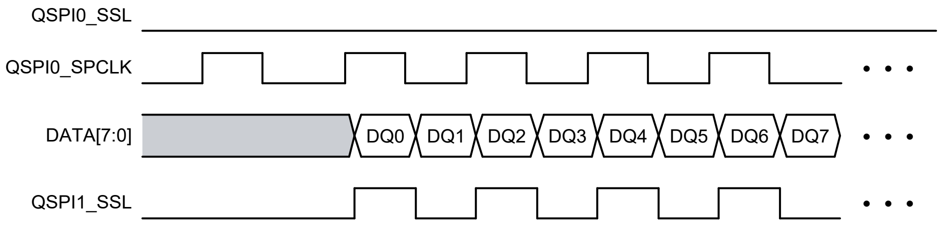
- The TEND bit is set to 1 on completion of the data transfer.

Figure 25.25 Procedure for Using the Write Buffer

25.5.14 Data Alignment for Octal-SPI Flash Memory

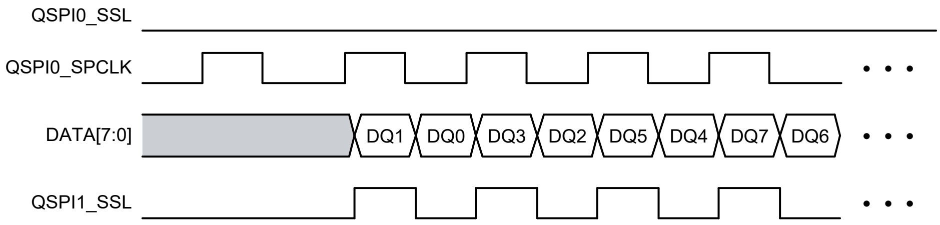
When the Octal-SPI flash memory is connected to this LSI, two data alignment types are available in this module.

Figure 25.26 shows the sequential alignment type and **Figure 25.27** shows the alternative alignment type.



Note: The DATA[7:0] lines are connected as 8-bit signals between this LSI and the Octal-SPI flash memory. (QSPI1_IO3 to QSPI0_IO0)

Figure 25.26 Octal-SPI Flash Memory Sequential Alignment (PHYCNT.OCTA = 10)



Note: The DATA[7:0] lines are connected as 8-bit signals between this LSI and the Octal-SPI flash memory. (QSPI1_IO3 to QSPI0_IO0)

Figure 25.27 Octal-SPI Flash Memory Alternative Alignment (PHYCNT.OCTA = 01)

25.5.15 Supported Protocol for Serial Flash Memory

The protocol for serial flash memory that this LSI supports is indicated in **Table 25.14**.

Table 25.14 Supported Protocol for Serial Flash Memory

Bit Width (Command-Address-Data)	Command	Address	Data
1-0-0	SDR	NA	NA
1-0-1	SDR	NA	SDR
1-1-1	SDR	SDR	SDR
	SDR	DDR	DDR
1-1-4	SDR	SDR	SDR
1-4-4	SDR	SDR	SDR
	SDR	DDR	DDR

25.5.16 Supported Protocol for Octal-SPI Flash Memory

The protocol for Octal-SPI flash memory that this LSI supports is indicated in **Table 25.15**.

Table 25.15 Supported Protocol for Octal-SPI Flash Memory

Bit Width (Command-Address-Data)	Command	Address	Data
1-0-0	SDR	NA	NA
1-0-1	SDR	NA	SDR
1-1-1	SDR	SDR	SDR
8-0-0	SDR	NA	NA
	DDR	NA	NA
8-0-8	DDR	NA	DDR
8-8-8	DDR	DDR	DDR

25.5.17 Timing Adjustment

When data is transferred with serial flash memory connected, adjustment of the timing between the clock operation and the acquisition of input data is required. When Octal-SPI flash memory or HyperFlash using data strobe connected, the acquisition timing of input data is automatically adjusted.

Figure 25.28 shows the timing adjustment flow for serial flash (SDR mode) connection.

Execute this flow before changing the setting of SCLKSEL.SPICR[1:0] from its initial value when the serial flash memory is connected (SDR mode). Execution of this flow at regular intervals is not required.

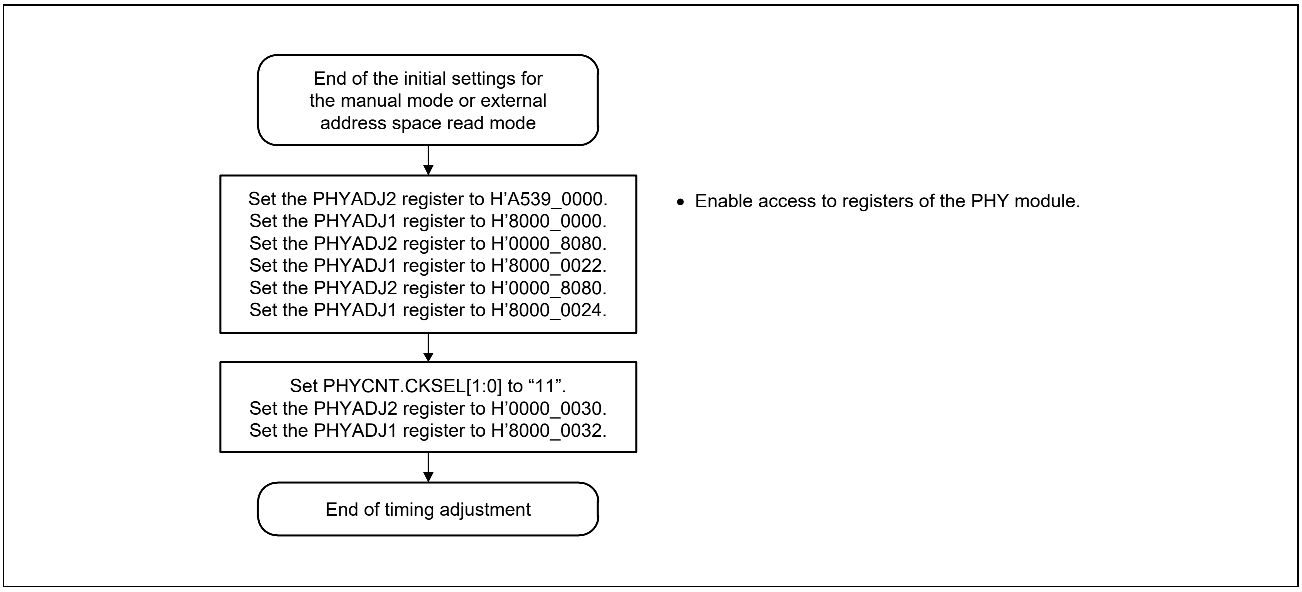
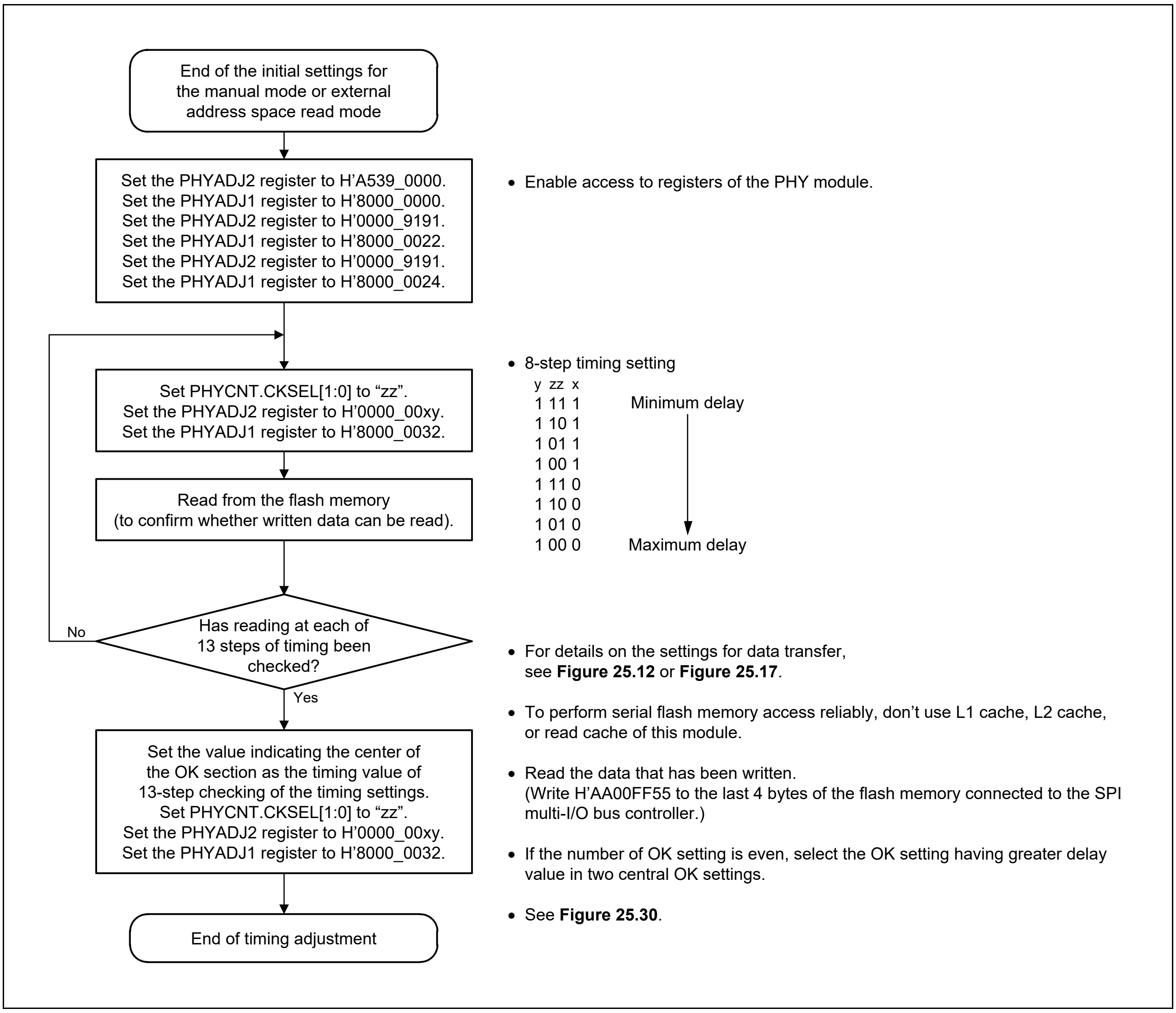


Figure 25.28 (1) Flow of Timing Adjustment when the Serial Flash Memory is Connected (SDR Mode)

Figure 25.29 shows the timing adjustment flow for serial flash (DDR mode) connection.

Since the set value acquired in this timing adjustment flow is affected by voltage and temperature variation, this flow should be performed periodically.



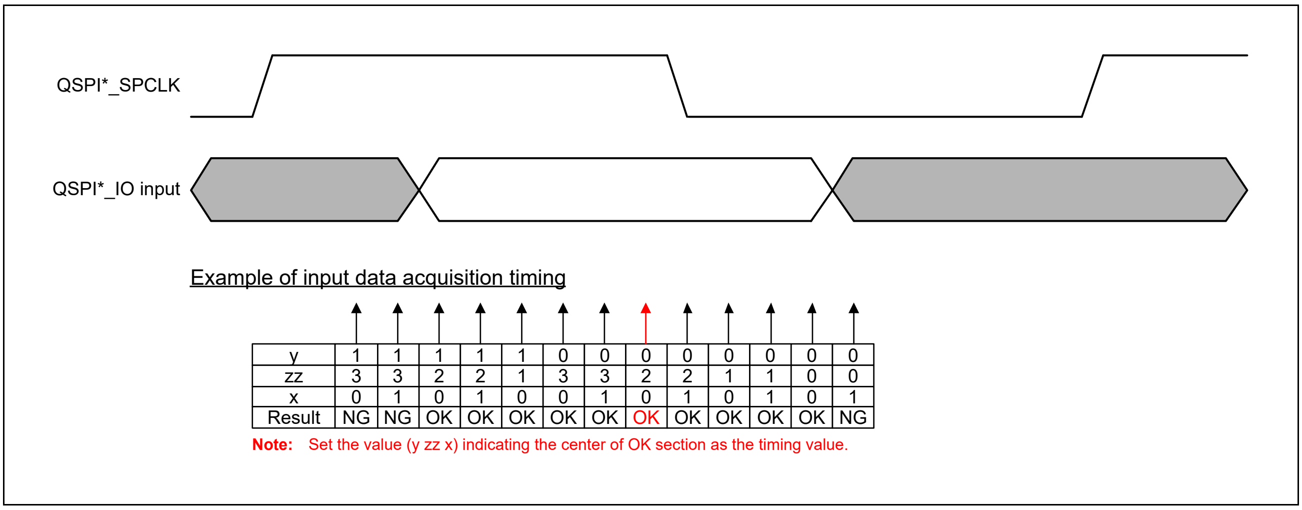
- Enable access to registers of the PHY module.
- 8-step timing setting

y	zz	x	
1	11	1	Minimum delay
1	10	1	
1	01	1	
1	00	1	
1	11	0	
1	10	0	
1	01	0	
1	00	0	Maximum delay

- For details on the settings for data transfer, see **Figure 25.12** or **Figure 25.17**.
- To perform serial flash memory access reliably, don't use L1 cache, L2 cache, or read cache of this module.
- Read the data that has been written. (Write H'AA00FF55 to the last 4 bytes of the flash memory connected to the SPI multi-I/O bus controller.)
- If the number of OK setting is even, select the OK setting having greater delay value in two central OK settings.
- See **Figure 25.30**.

Figure 25.29 (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode)

Figure 25.30 shows an example of timing setting for serial flash (DDR mode) connection.



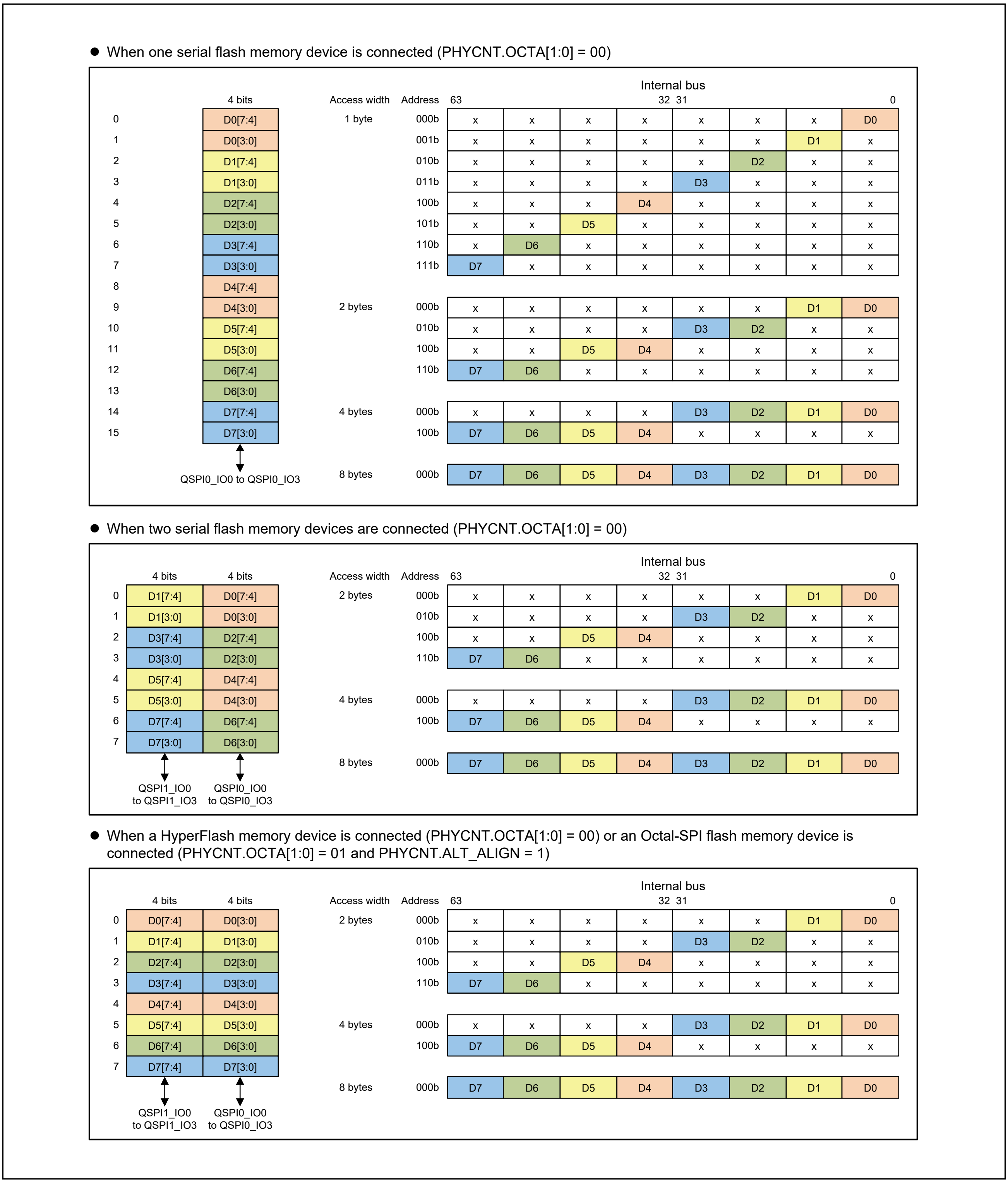
y	1	1	1	1	1	0	0	0	0	0	0	0	0
zz	3	3	2	2	1	3	3	2	2	1	1	0	0
x	0	1	0	1	0	0	1	0	1	0	1	0	1
Result	NG	NG	OK	OK	OK	OK	OK	OK	OK	OK	OK	OK	NG

Note: Set the value (y zz x) indicating the center of OK section as the timing value.

Figure 25.30 (3) Example of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode)

25.5.18 Data Alignment

Data alignments in external address space read mode and Manual mode are shown in **Figure 25.31** and **Figure 25.32**, respectively. When two serial flash memory devices are connected, the addresses of one device connected to pins QSPI0_IO0 to QSPI0_IO3 are treated as 2n and those of the other device connected to pins QSPI1_IO0 to QSPI1_IO3 are treated as 2n + 1. Be sure that access is in at least word units. Access in byte units is not supported.



- When one serial flash memory device is connected (PHYCNT.OCTA[1:0] = 00)

	4 bits
0	D0[7:4]
1	D0[3:0]
2	D1[7:4]
3	D1[3:0]
4	D2[7:4]
5	D2[3:0]
6	D3[7:4]
7	D3[3:0]
8	D4[7:4]
9	D4[3:0]
10	D5[7:4]
11	D5[3:0]
12	D6[7:4]
13	D6[3:0]
14	D7[7:4]
15	D7[3:0]

QSPI0_IO0 to QSPI0_IO3

Internal bus

Access width	Address	63			32	31			0
1 byte	000b	x	x	x	x	x	x	x	D0
	001b	x	x	x	x	x	x	D1	x
	010b	x	x	x	x	x	D2	x	x
	011b	x	x	x	x	D3	x	x	x
	100b	x	x	x	D4	x	x	x	x
	101b	x	x	D5	x	x	x	x	x
	110b	x	D6	x	x	x	x	x	x
	111b	D7	x	x	x	x	x	x	x
2 bytes	000b	x	x	x	x	x	x	D1	D0
	010b	x	x	x	x	D3	D2	x	x
	100b	x	x	D5	D4	x	x	x	x
	110b	D7	D6	x	x	x	x	x	x
4 bytes	000b	x	x	x	x	D3	D2	D1	D0
	100b	D7	D6	D5	D4	x	x	x	x
8 bytes	000b	D7	D6	D5	D4	D3	D2	D1	D0

- When two serial flash memory devices are connected (PHYCNT.OCTA[1:0] = 00)

	4 bits	4 bits
0	D1[7:4]	D0[7:4]
1	D1[3:0]	D0[3:0]
2	D3[7:4]	D2[7:4]
3	D3[3:0]	D2[3:0]
4	D5[7:4]	D4[7:4]
5	D5[3:0]	D4[3:0]
6	D7[7:4]	D6[7:4]
7	D7[3:0]	D6[3:0]
	QSPI1_IO0 to QSPI1_IO3	QSPI0_IO0 to QSPI0_IO3

Internal bus

Access width	Address	63			32	31			0
2 bytes	000b	x	x	x	x	x	x	D1	D0
	010b	x	x	x	x	D3	D2	x	x
	100b	x	x	D5	D4	x	x	x	x
	110b	D7	D6	x	x	x	x	x	x
4 bytes	000b	x	x	x	x	D3	D2	D1	D0
	100b	D7	D6	D5	D4	x	x	x	x
8 bytes	000b	D7	D6	D5	D4	D3	D2	D1	D0

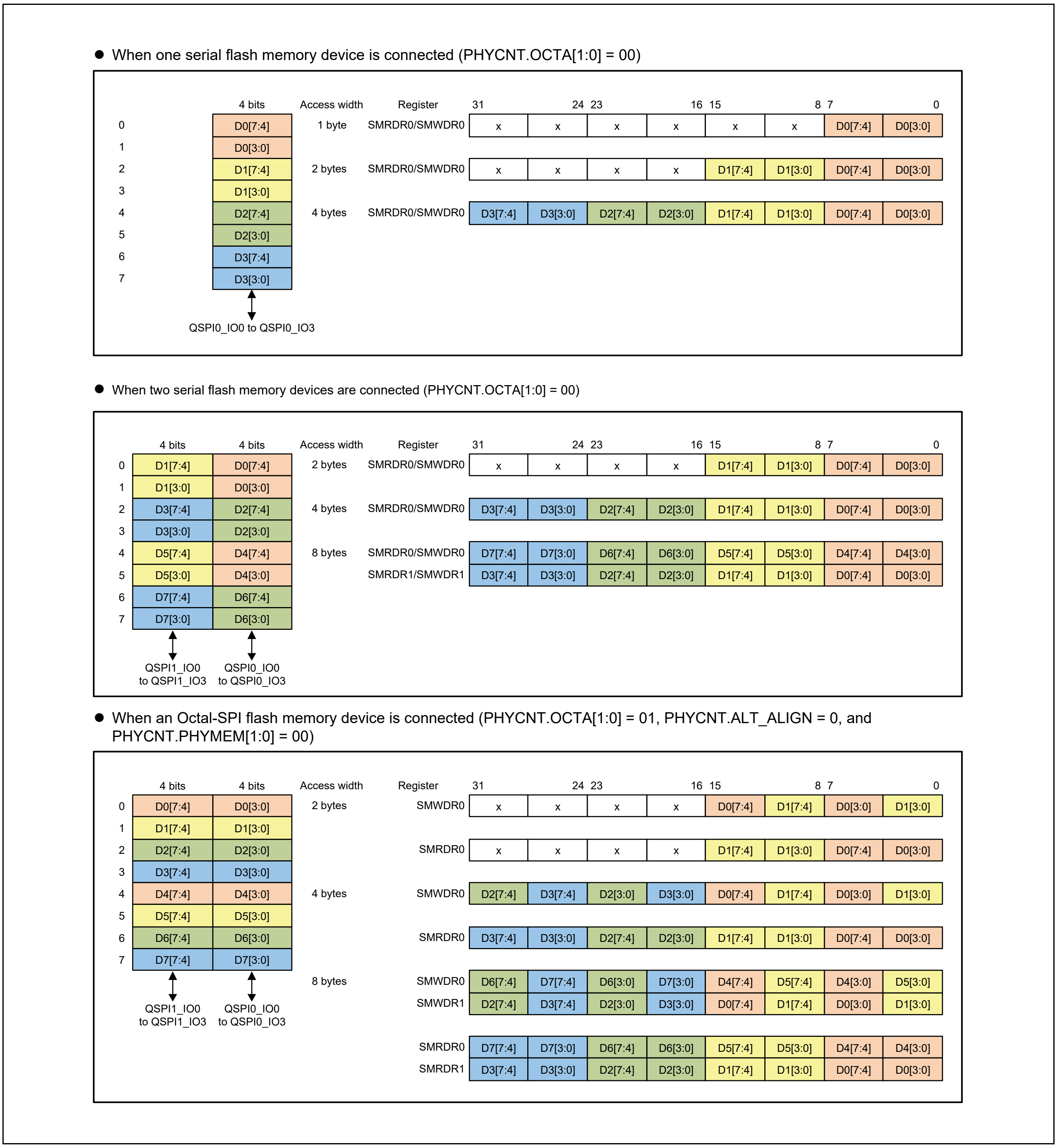
- When a HyperFlash memory device is connected (PHYCNT.OCTA[1:0] = 00) or an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01 and PHYCNT.ALT_ALIGN = 1)

	4 bits	4 bits
0	D0[7:4]	D0[3:0]
1	D1[7:4]	D1[3:0]
2	D2[7:4]	D2[3:0]
3	D3[7:4]	D3[3:0]
4	D4[7:4]	D4[3:0]
5	D5[7:4]	D5[3:0]
6	D6[7:4]	D6[3:0]
7	D7[7:4]	D7[3:0]
	QSPI1_IO0 to QSPI1_IO3	QSPI0_IO0 to QSPI0_IO3

Internal bus

Access width	Address	63			32	31			0
2 bytes	000b	x	x	x	x	x	x	D1	D0
	010b	x	x	x	x	D3	D2	x	x
	100b	x	x	D5	D4	x	x	x	x
	110b	D7	D6	x	x	x	x	x	x
4 bytes	000b	x	x	x	x	D3	D2	D1	D0
	100b	D7	D6	D5	D4	x	x	x	x
8 bytes	000b	D7	D6	D5	D4	D3	D2	D1	D0

Figure 25.31 Data Alignment in External Address Space Read Mode



- When one serial flash memory device is connected (PHYCNT.OCTA[1:0] = 00)

	4 bits
0	D0[7:4]
1	D0[3:0]
2	D1[7:4]
3	D1[3:0]
4	D2[7:4]
5	D2[3:0]
6	D3[7:4]
7	D3[3:0]

QSPI0_IO0 to QSPI0_IO3

Access width	Register	31		24 23		16 15		8 7	0
1 byte	SMRDR0/SMWDR0	x	x	x	x	x	x	D0[7:4]	D0[3:0]
2 bytes	SMRDR0/SMWDR0	x	x	x	x	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
4 bytes	SMRDR0/SMWDR0	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]

- When two serial flash memory devices are connected (PHYCNT.OCTA[1:0] = 00)

	4 bits	4 bits
0	D1[7:4]	D0[7:4]
1	D1[3:0]	D0[3:0]
2	D3[7:4]	D2[7:4]
3	D3[3:0]	D2[3:0]
4	D5[7:4]	D4[7:4]
5	D5[3:0]	D4[3:0]
6	D7[7:4]	D6[7:4]
7	D7[3:0]	D6[3:0]
	QSPI1_IO0 to QSPI1_IO3	QSPI0_IO0 to QSPI0_IO3

Access width	Register	31		24 23		16 15		8 7	0
2 bytes	SMRDR0/SMWDR0	x	x	x	x	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
4 bytes	SMRDR0/SMWDR0	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
8 bytes	SMRDR0/SMWDR0	D7[7:4]	D7[3:0]	D6[7:4]	D6[3:0]	D5[7:4]	D5[3:0]	D4[7:4]	D4[3:0]
	SMRDR1/SMWDR1	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]

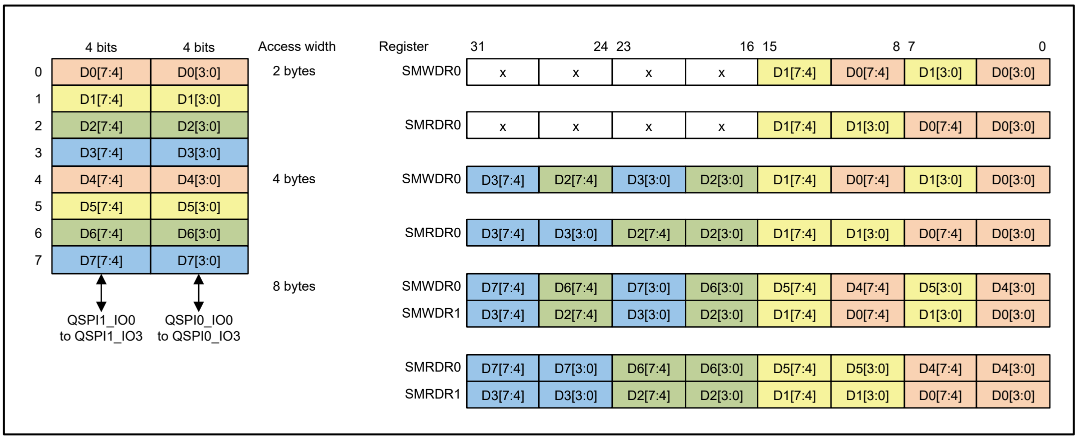
- When an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01, PHYCNT.ALT_ALIGN = 0, and PHYCNT.PHYMEM[1:0] = 00)

	4 bits	4 bits
0	D0[7:4]	D0[3:0]
1	D1[7:4]	D1[3:0]
2	D2[7:4]	D2[3:0]
3	D3[7:4]	D3[3:0]
4	D4[7:4]	D4[3:0]
5	D5[7:4]	D5[3:0]
6	D6[7:4]	D6[3:0]
7	D7[7:4]	D7[3:0]
	QSPI1_IO0 to QSPI1_IO3	QSPI0_IO0 to QSPI0_IO3

Access width	Register	31		24 23		16 15		8 7	0
2 bytes	SMWDR0	x	x	x	x	D0[7:4]	D1[7:4]	D0[3:0]	D1[3:0]
	SMRDR0	x	x	x	x	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
4 bytes	SMWDR0	D2[7:4]	D3[7:4]	D2[3:0]	D3[3:0]	D0[7:4]	D1[7:4]	D0[3:0]	D1[3:0]
	SMRDR0	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
8 bytes	SMWDR0	D6[7:4]	D7[7:4]	D6[3:0]	D7[3:0]	D4[7:4]	D5[7:4]	D4[3:0]	D5[3:0]
	SMWDR1	D2[7:4]	D3[7:4]	D2[3:0]	D3[3:0]	D0[7:4]	D1[7:4]	D0[3:0]	D1[3:0]
	SMRDR0	D7[7:4]	D7[3:0]	D6[7:4]	D6[3:0]	D5[7:4]	D5[3:0]	D4[7:4]	D4[3:0]
	SMRDR1	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]

Figure 25.32 Data Alignment in Manual Mode (1/2)

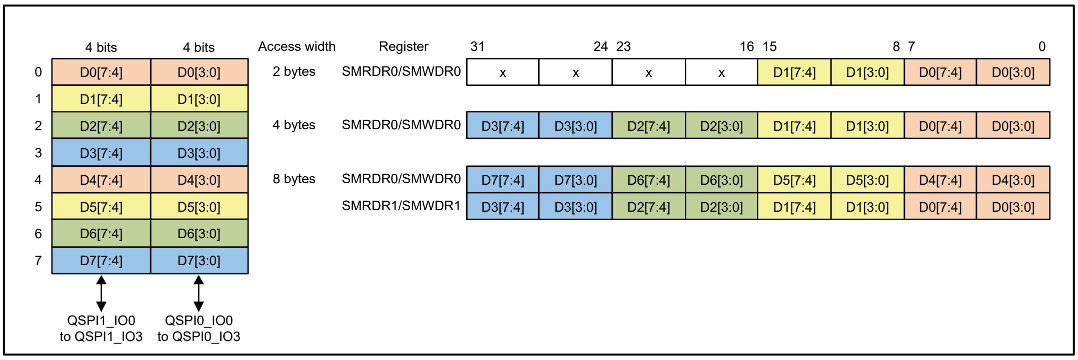
- When an Octal-SPI flash memory device is connected (PHYCNT.OCTA[1:0] = 01, PHYCNT.ALT_ALIGN = 1, and PHYCNT.PHYMEM[1:0] = 01)



	4 bits	4 bits
0	D0[7:4]	D0[3:0]
1	D1[7:4]	D1[3:0]
2	D2[7:4]	D2[3:0]
3	D3[7:4]	D3[3:0]
4	D4[7:4]	D4[3:0]
5	D5[7:4]	D5[3:0]
6	D6[7:4]	D6[3:0]
7	D7[7:4]	D7[3:0]

Access width	Register	31	24	23	16	15	8	7	0
2 bytes	SMWDR0	x	x	x	x	D1[7:4]	D0[7:4]	D1[3:0]	D0[3:0]
	SMRDR0	x	x	x	x	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
4 bytes	SMWDR0	D3[7:4]	D2[7:4]	D3[3:0]	D2[3:0]	D1[7:4]	D0[7:4]	D1[3:0]	D0[3:0]
	SMRDR0	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
8 bytes	SMWDR0	D7[7:4]	D6[7:4]	D7[3:0]	D6[3:0]	D5[7:4]	D4[7:4]	D5[3:0]	D4[3:0]
	SMWDR1	D3[7:4]	D2[7:4]	D3[3:0]	D2[3:0]	D1[7:4]	D0[7:4]	D1[3:0]	D0[3:0]
	SMRDR0	D7[7:4]	D7[3:0]	D6[7:4]	D6[3:0]	D5[7:4]	D5[3:0]	D4[7:4]	D4[3:0]
	SMRDR1	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]

- When a HyperFlash memory device is connected (PHYCNT.OCTA[1:0] = 00)



	4 bits	4 bits
0	D0[7:4]	D0[3:0]
1	D1[7:4]	D1[3:0]
2	D2[7:4]	D2[3:0]
3	D3[7:4]	D3[3:0]
4	D4[7:4]	D4[3:0]
5	D5[7:4]	D5[3:0]
6	D6[7:4]	D6[3:0]
7	D7[7:4]	D7[3:0]

Access width	Register	31	24	23	16	15	8	7	0
2 bytes	SMRDR0/SMWDR0	x	x	x	x	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
4 bytes	SMRDR0/SMWDR0	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]
8 bytes	SMRDR0/SMWDR0	D7[7:4]	D7[3:0]	D6[7:4]	D6[3:0]	D5[7:4]	D5[3:0]	D4[7:4]	D4[3:0]
	SMRDR1/SMWDR1	D3[7:4]	D3[3:0]	D2[7:4]	D2[3:0]	D1[7:4]	D1[3:0]	D0[7:4]	D0[3:0]

Figure 25.32 Data Alignment in Manual Mode (2/2)

25.6 Usage Notes

25.6.1 Transfer to read data while the signal on the QSPIn_SSL pin is de-asserted

Set the SMENR.SPIDE[3:0] bits to 1100 or 1111 when transfer only for reading data is to proceed.

Transfer will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000.

25.6.2 Transfer to read data while the signal on the QSPIn_SSL pin is asserted

When transfer only for reading data is to proceed, set the SMENR.SPIDE[3:0] bits to 1100 or 1111, or end the immediately preceding transfer with reading data.

When the immediately preceding transfer is of a command, optional command, address, or option data, or is transfer for writing data, the subsequent transfer only for reading data will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000.

25.6.3 Notes on Starting Transfer from the QSPIn_SSL Retained State in Manual Mode

Be sure to set the SPIWE bit in the SMCR register to 1 when the transfer of a command, optional command, address, or option data is started while the QSPIn_SSL pin is being asserted in Manual mode.

In addition, while the QSPIn_SSL signal is being asserted in Manual mode, the setting to start data transfer for reading is prohibited.

25.6.4 Handling of QSPI_RESET#

The state of the QSPI_RESET# pin can be changed in the following two ways.

1. Use the RSTEN and RSTVAL bits in the PHYINT register.
   The state of the QSPI_RESET# pin depends on the value of the RETVAL bits in the PHYINT register.
2. Activate the reset to this module.
   While a power-on reset or a software reset is applied to this LSI, the signal on the QSPI_RESET# pin goes low.
   When the reset is deactivated, the signal on the QSPI_RESET# pin goes high.

25.6.5 Software Reset after Release from the Module Stop State

When resetting SPI Multi I/O using the CPG reset ON/OFF register, set the clock ON/OFF register to ON before releasing the reset to supply the clock. While the SPI Multi I/O is in the reset state, the state of the QSPI_RESET# pin is L. Guarantee the reset period and accessible period after reset required by the connected device.

25.6.6 Write Data when the Octal-SPI Flash Memory is Connected

Control the alignment of the write data appropriately when writing data to the Octal-SPI flash memory connected to this LSI.

For details, refer to **Section 25.4.15, Manual Mode Write Data Register 0 (SMWDR0)**, and **Section 25.4.16, Manual Mode Write Data Register 1 (SMWDR1)**.

26. I^{2}C Bus Interface

This section gives an overall description of the I^{2}C bus interface (RIIC).

The first section describes the features specific to this LSI, including the number of units and the register base addresses. The subsequent sections describe the RIIC's functions and registers.

26.1 Features

26.1.1 Channels

This LSI has the following number of channels of the I^{2}C bus interface (RIIC).

Table 26.1 Channels of RIIC

Item	Description
Number of channels	4
Name	RIICn (n = 0 to 3)

Table 26.2 Index

Index	Description
n	Throughout this section, the individual channels of the I^{2}C bus interface are identified by the index "n" (n = 0 to 3); for example, RIICnCR1 for the I^{2}C bus control register 1.

26.1.2 Register Base Addresses

The base address <RIICn_base> of each RIICn is listed in the following table.

All RIICn register addresses are given as values obtained by adding offsets to the register base address <RIICn_base> for each channel.

Table 26.3 Register Base Address

Channel	Base Address Name	Base Address
RIIC0	<RIIC0_base>	H'0_1005_8000 (Cortex-A55 Address Space)
RIIC1	<RIIC1_base>	H'0_1005_8400 (Cortex-A55 Address Space)
RIIC2	<RIIC2_base>	H'0_1005_8800 (Cortex-A55 Address Space)
RIIC3	<RIIC3_base>	H'0_1005_8C00 (Cortex-A55 Address Space)
RIIC0	<RIIC0_base>	H'4005_8000 (Cortex-M33 Address Space Non-Secure)
RIIC1	<RIIC1_base>	H'4005_8400 (Cortex-M33 Address Space Non-Secure)
RIIC2	<RIIC2_base>	H'4005_8800 (Cortex-M33 Address Space Non-Secure)
RIIC3	<RIIC3_base>	H'4005_8C00 (Cortex-M33 Address Space Non-Secure)
RIIC0	<RIIC0_base>	H'5005_8000 (Cortex-M33 Address Space Secure)
RIIC1	<RIIC1_base>	H'5005_8400 (Cortex-M33 Address Space Secure)
RIIC2	<RIIC2_base>	H'5005_8800 (Cortex-M33 Address Space Secure)
RIIC3	<RIIC3_base>	H'5005_8C00 (Cortex-M33 Address Space Secure)

26.1.3 External I/O Signals

The following table shows the external I/O signals of the RIIC.

Table 26.4 RIICn Pin Configuration

Channel	Alternative Port Pin Name	Function
RIIC0	RIIC0_SCL	RIIC0 serial clock I/O pin
	RIIC0_SDA	RIIC0 serial data I/O pin
RIIC1	RIIC1_SCL	RIIC1 serial clock I/O pin
	RIIC1_SDA	RIIC1 serial data I/O pin
RIIC2	RIIC2_SCL	RIIC2 serial clock I/O pin
	RIIC2_SDA	RIIC2 serial data I/O pin
RIIC3	RIIC3_SCL	RIIC3 serial clock I/O pin
	RIIC3_SDA	RIIC3 serial data I/O pin

Table 26.5 I2C Bus Interface

Channel Number*[3]	Pin Name*[1]	I/O	Function	This LSI
				Output Buffer Type*[2]
0	SCL	I/O	SCL: I2C serial clock input/output pin SDA: I2C serial data input/output pin	OD
	SDA			
1	SCL			OD
	SDA			
2	SCL			LVTTL
	SDA			
3	SCL			LVTTL
	SDA			

Note 1. The actual pin names are RIICnSCL and RIICnSDA.

Note 2. Output buffer type: “OD” is open drain buffer and “LVTTL” is low level drive only LVTTL buffer.

Note 3. AC specification is different between channel 0, 1 and channel 2, 3.

26.2 Overview

26.2.1 Functional Overview

Communications format

- I^{2}C bus format or SMBus format
- Master mode or slave mode selectable
- Automatic securing of the various set-up times, hold times, and bus-free times for the transfer rate

Transfer rate

Up to 1 Mbps

SCL clock

For master operation, the duty cycle of the SCL clock is selectable in the following range. 0% < Duty cycle < 100%.

Issuing and detecting conditions

- Start, restart, and stop conditions are automatically generated.
- Start conditions (including restart conditions) and stop conditions are detected.

Slave address

- Up to three slave-address settings can be made.
- Seven- and ten-bit address formats are supported (along with the use of both at once).
- General call addresses, device ID addresses, and SMBus host addresses are detected.

Acknowledgement

- For transmission, the acknowledge bit is automatically loaded
  - Transfer of the next data for transmission can be automatically suspended on detection of a not-acknowledge bit.
- For reception, the acknowledge bit is automatically transmitted
  - If a wait between the eighth and ninth clock cycles has been selected, software control of the value in the acknowledge field in response to the received value is possible.

Wait function

- In reception, the following periods of waiting can be obtained by holding the clock signal (SCL) at the low level:
  - Waiting between the eighth and ninth clock cycles
  - Waiting between the ninth clock cycle and the first clock cycle of the next transfer (WAIT function)

SDA output delay function

Timing of the output of transmitted data, including the acknowledge bit, can be delayed.

Arbitration

- For multi-master operation
  - Operation to synchronize the SCL (clock) signal in cases of conflict with the SCL signal from another master is possible.
  - When issuing the start condition would create conflict on the bus, loss of arbitration is detected by testing for non-matching between the internal signal for the SDA line and the level on the SDA line.
  - In master operation, loss of arbitration is detected by testing for non-matching between the signal on the SDA line and the internal signal for the SDA line.
- Loss of arbitration due to detection of the start condition while the bus is busy is detectable (to prevent the issuing of double start conditions).
- Loss of arbitration in transfer of a not-acknowledge bit due to the internal signal for the SDA line and the level on the SDA line not matching is detectable.
- Loss of arbitration due to non-matching of internal and line levels for data is detectable in slave transmission.

Timeout function

The internal time-out function is capable of detecting long-interval stop of the SCL (clock signal).

Noise removal

The interface incorporates analog noise filters and digital noise filters for input on the RIICnSCL and RIICnSDA pins, and the width for noise cancellation by the digital noise filters is adjustable by software.

Interrupt sources

- Eight sources:
  - Transmission complete
  - Receive-data-full
  - Transmit-data-empty
  - Detection of a stop condition
  - Detection of a start condition
  - Reception of a NACK
  - Arbitration lost
  - Timeout

Low power consumption function

Module-stop state can be set.

26.2.2 Block Diagram

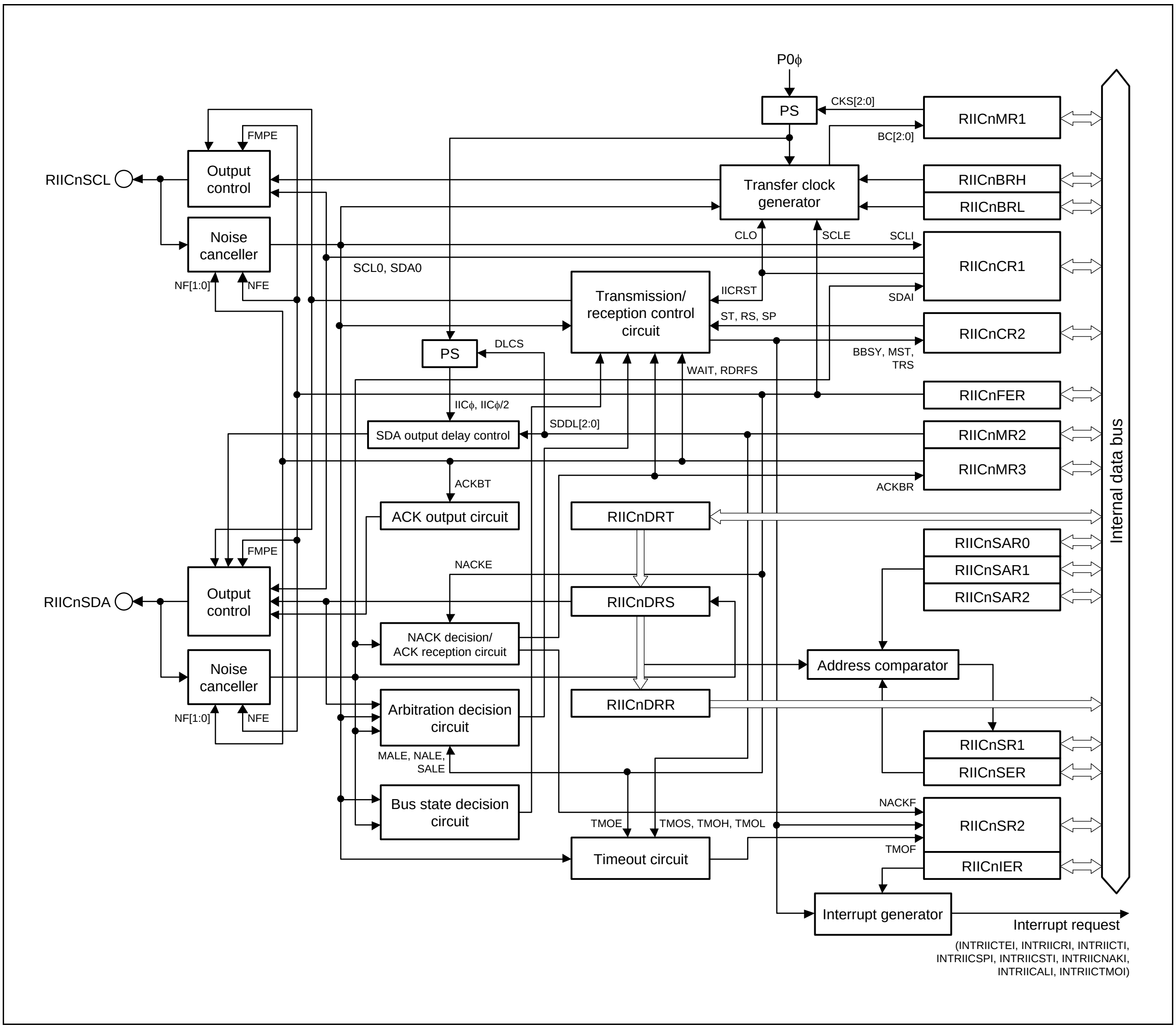


Figure 26.1 Block Diagram of RIIC

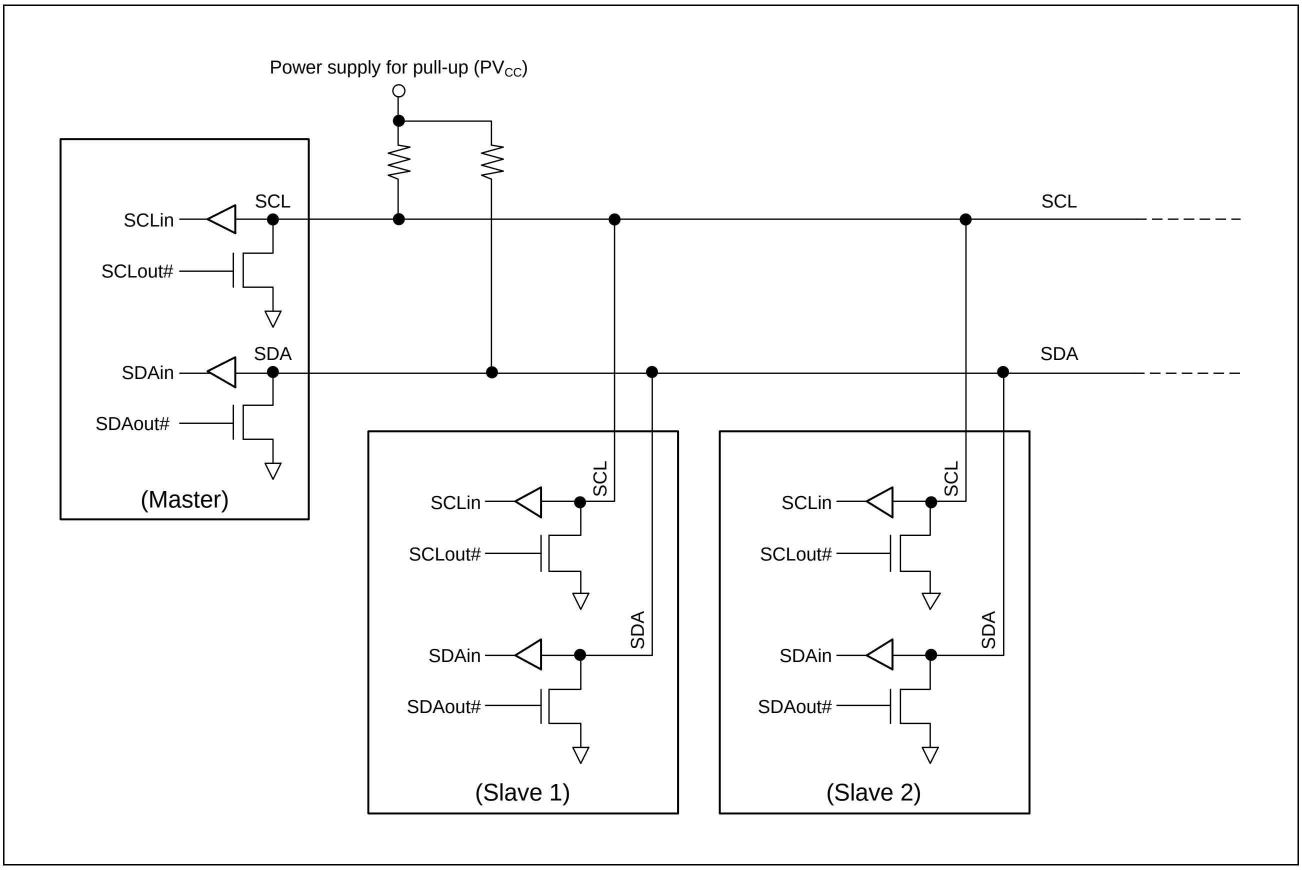


Figure 26.2 Connections to the External Circuit by the I/O Pins (I^{2}C Bus Configuration Example)

RIICnSCL and RIICnSDA are Schmitt input/open-drain output pins for both master and slave operations. Because the output is open drain, an external pull-up resistor is required.

26.3 Registers

Channel	Register Name	Abbreviation	Offset Address	Access Size
RIICn (n = 0 to 3)	I^{2}C bus control register 1	ICCR1	H’00	8, 16, 32
	I^{2}C bus control register 2	ICCR2	H’04	8, 16, 32
	I^{2}C bus mode register 1	ICMR1	H’08	8, 16, 32
	I^{2}C bus mode register 2	ICMR2	H’0C	8, 16, 32
	I^{2}C bus mode register 3	ICMR3	H’10	8, 16, 32
	I^{2}C bus function enable register	ICFER	H’14	8, 16, 32
	I^{2}C bus status enable register	ICSER	H’18	8, 16, 32
	I^{2}C bus interrupt enable register	ICIER	H’1C	8, 16, 32
	I^{2}C bus status register 1	ICSR1	H’20	8, 16, 32
	I^{2}C bus status register 2	ICSR2	H’24	8, 16, 32
	I^{2}C slave address register 0	ICSAR0	H’28	8, 16, 32
	I^{2}C slave address register 1	ICSAR1	H’2C	8, 16, 32
	I^{2}C slave address register 2	ICSAR2	H’30	8, 16, 32
	I^{2}C bus bit rate low-level register	ICBRL	H’34	8, 16, 32
	I^{2}C bus bit rate high-level register	ICBRH	H’38	8, 16, 32
	I^{2}C bus transmit data register	ICDRT	H’3C	8, 16, 32
	I^{2}C bus receive data register	ICDRR	H’40	8, 16, 32

26.3.1 RIICnCR1 — I^{2}C Bus Control Register 1

Access Size: RIICnCR1 is a 32-bit readable/writable register.
RIICnCR1L and RIICnCR1H are 16-bit readable/writable registers.
RIICnCR1LL, RIICnCR1LH, RIICnCR1HL, and RIICnCR1HH are 8-bit readable/writable registers.

Address(es): RIICnCR1: <RIICn_base> + H’0000
RIICnCR1L: <RIICn_base> + H’0000, RIICnCR1H: <RIICn_base> + H’0002
RIICnCR1LL: <RIICn_base> + H’0000, RIICnCR1LH: <RIICn_base> + H’0001,
RIICnCR1HL: <RIICn_base> + H’0002, RIICnCR1HH: <RIICn_base> + H’0003

Initial Value: H’0000 001F. This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	ICE	0	R/W	I^{2}C Bus Interface Enable 0: Output to the RIICnSCL and RIICnSDA pins is disabled. (Input to the RIICnSCL and RIICnSDA pins is enabled.) 1: Enabled (RIICnSCL and RIICnSDA pins are in the driving state) (An RIIC reset or an internal reset is selected according to the combination of this bit and IICRST bit settings.)
6	IICRST	0	R/W	I^{2}C Bus Interface Internal Reset 0: Clears the RIIC reset or internal reset. 1: Initiates the RIIC reset or internal reset. (Clears the bit counter and the SCL/SDA output latch)
5	CLO	0	R/W	Extra SCL Clock Cycle Output 0: Does not output an extra SCL clock cycle (default). 1: Outputs an extra SCL clock cycle. (The CLO bit is cleared automatically after one clock cycle is output.)
4	SOWP*[2]	1	R/W	SCLO/SDAO Write Protect 0: Allows the SCLO and SDAO bits to be rewritten. (This bit is read as 1.)
3	SCLO*[1, *2]	1	R/W	SCL Output Control • Read: 0: RIICnSCL pin output is at a low level. 1: RIICnSCL pin is in a high-impedance state. • Write: 0: Changes the RIICnSCL pin output to a low level. 1: Changes the RIICnSCL pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)

Bit	Bit Name	Initial Value	R/W	Description
2	SDAO[*1,*2]	1	R/W	SDA Output Control ● Read: 0: RIICnSDA pin output is at a low level. 1: RIICnSDA pin is in a high-impedance state. ● Write: 0: Changes the RIICnSDA pin output to a low level. 1: Changes the RIICnSDA pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)
1	SCLI	1	R	SCL Bus Input Monitor 0: RIICnSCL pin input is at a low level. 1: RIICnSCL pin input is at a high level.
0	SDAI	1	R	SDA Bus Input Monitor 0: RIICnSDA pin input is at a low level. 1: RIICnSDA pin input is at a high level.

Note 1. Do not write to these bits during communication. Changing a value during communication may cause a transmission or reception failure or an AL error.

Note 2. To change the SDAO and SCLO bits, set the SOWP bit to 0 at the same timing to set the SDAO and SCLO bits to 0.

CLO Bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.

Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error. For details on this function, see **Section 26.13.2, Extra SCL Clock Cycle Output Function**.

IICRST Bit (I^{2}C Bus Interface Internal Reset)

This bit is used to reset the internal states of the RIIC. Setting this bit to 1 initiates an RIIC reset or internal reset.

Whether an RIIC reset or internal reset is initiated is determined according to the combination with the ICE bit. **Table 26.6** lists the resets of the RIIC.

The RIIC reset resets all registers including the RIICnCR2.BBSY flag (except ICE and IICRST) and internal states of the RIIC, and the internal reset resets the bit counter (RIICnMR1.BC[2:0] bits), the I^{2}C bus shift register (RIICnDRS), and the I^{2}C bus status registers (RIICnSR1 and RIICnSR2) as well as the internal states of the RIIC. For the reset conditions for each register, see **Section 26.15, Reset Function of RIIC**.

An internal reset initiated with the IICRST bit set to 1 during operation (with the ICE bit set to 1) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up due to a communication error.

If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the RIICnSCL pin and RIICnSDA pin at a high impedance.

CAUTION

If an internal reset is initiated using the IICRST bit for a bus hang-up occurred during communication with the master device in slave mode, the states may become different between the slave device and the master device (due to the difference in the bit counter information). For this reason, do not initiate an internal reset in slave mode, but initiate restoration processing from the master device. If an internal reset is necessary because the RIIC hangs up with the SCL line in a low level output state in slave mode, initiate an internal reset and then issue a restart condition from the master device or resume communication from the start condition issuance after issuing a stop condition. If communication is restarted by initiating a reset solely in the slave device without issuing a start condition or restart condition from the master device, synchronization will be lost because the master and slave devices operate asynchronously.

Table 26.6 RIIC Resets

IICRST	ICE	State	Specifications
1	0	RIIC reset	Resets all registers (except ICE and IICRST) and internal states of the RIIC.
	1	Internal reset	Reset the RIICnMR1.BC[2:0] bits, and the RIICnSR1, RIICnSR2, RIICnDRS registers and the internal states of the RIIC.

ICE Bit (I^{2}C Bus Interface Enable)

The ICE bit selects driving or non-driving of the RIICnSCL and RIICnSDA pins. Moreover, this bit can perform two types of reset in combination with the IICRST bit. For the types of reset, see **Table 26.6**.

Set the ICE bit to 1 when using RIIC. Setting the ICE bit to 1 selects driving of the RIICnSCL and RIICnSDA pins. Set the ICE bit to 0 when RIIC is not to be used. Clearing the ICE bit to 0 disables output from the RIICnSCL and RIICnSDA pins.

26.3.2 RIICnCR2 — I^{2}C Bus Control Register 2

Access Size:	RIICnCR2 is a 32-bit readable/writable register. RIICnCR2L and RIICnCR2H are 16-bit readable/writable registers. RIICnCR2LL, RIICnCR2LH, RIICnCR2HL, and RIICnCR2HH are 8-bit readable/writable registers.
Address(es):	RIICnCR2: <RIICn_base> + H’0004 RIICnCR2L: <RIICn_base> + H’0004, RIICnCR2H: <RIICn_base> + H’0006 RIICnCR2LL: <RIICn_base> + H’0004, RIICnCR2LH: <RIICn_base> + H’0005, RIICnCR2HL: <RIICn_base> + H’0006, RIICnCR2HH: <RIICn_base> + H’0007
Initial Value:	H’0000 0000 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BBSY	MST	TRS	—	SP	RS	ST	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	BBSY	0	R	Bus Busy Detection Flag 0: The I^{2}C bus is released (bus free state). 1: The I^{2}C bus is occupied (bus busy state or in the bus free state).
6	MST	0	R	Master/Slave Mode 0: Slave mode 1: Transmit mode
5	TRS	0	R	Transmit/Receive Mode 0: Receive mode 1: Transmit mode
4	—	0	R	Reserved These bits are read as 0. The write value should be 0.
3	SP	0	R/W	Stop Condition Issuance Request 0: Does not request to issue a stop condition. 1: Requests to issue a stop condition.
2	RS	0	R/W	Restart Condition Issuance Request 0: Does not request to issue a restart condition. 1: Requests to issue a restart condition.
1	ST	0	R/W	Start Condition Issuance Request 0: Does not request to issue a start condition. 1: Requests to issue a start condition.
0	—	0	R	Reserved These bits are read as 0. The write value should be 0.

ST Bit (Start Condition Issuance Request)

This bit is used to request transition to master mode and issuance of a start condition.

When this bit is set to 1 to request to issue a start condition, a start condition is issued when the BBSY flag is set to 0 (bus free).

For details on the start condition issuance, see **Section 26.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the ST bit

[Clearing conditions]

- When 0 is written to the ST bit
- When a start condition has been issued
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

CAUTION

Set the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 0 (bus free).

Note that arbitration may be lost if the ST bit is set to 1 (start condition issuance request) when the BBSY flag is set to 1 (bus busy).

RS Bit (Restart Condition Issuance Request)

This bit is used to request that a restart condition be issued in master mode.

When this bit is set to 1 to request to issue a restart condition, a restart condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).

For details on the restart condition issuance, see **Section 26.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the RS bit with the RIICnCR2.BBSY flag set to 1

[Clearing conditions]

- When 0 is written to the RS bit
- When a restart condition has been issued or a start condition is detected
- When a stop condition is detected
- When the RIICnCR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

CAUTIONS

1. Do not set the RS bit to 1 while issuing a stop condition.
2. It is commended to issue a restart condition in master transmit mode. If the RS bit is set to 1 (restart condition issuance request) in mode other than master mode, the restart condition is not issued in this mode but the restart condition issuance request bit remains set. If the operating mode changes to master mode with the bit not being cleared, the restart condition may be issued.

SP Bit (Stop Condition Issuance Request)

This bit is used to request that a stop condition be issued in master mode.

When this bit is set to 1 to request to issue a stop condition, a stop condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).

For details on the stop condition issuance, see **Section 26.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the SP bit with both the RIICnCR2.BBSY flag and the RIICnCR2.MST bit set to 1

[Clearing conditions]

- When 0 is written to the SP bit
- When a stop condition has been issued or a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When a start condition and a restart condition are detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

CAUTIONS

1. Wring to the SP bit is not possible while the setting of the BBSY flag is 0 (bus free).
2. Do not set the SP bit to 1 while a restart condition is being issued.

TRS Bit (Transmit/Receive Mode)

This bit indicates transmit or receive mode.

The RIIC is in receive mode when the TRS bit is set to 0 and is in transmit mode when the bit is set to 1. Combination of this bit and the MST bit indicates the operating mode of the RIIC.

The value of the TRS bit is automatically changed to the value for transmission mode or reception mode by detection or issuing of a start condition, setting or clearing of the R/W# bit, etc.

[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)
- When the R/W# bit added to the slave address is set to 0 in master mode
- When the address received in slave mode matches the address enabled in RIICnSER, with the R/W# bit set to 1

[Clearing conditions]

- When a stop condition is detected
- The RIICnSR2.AL (arbitration-lost) flag being set to 1
- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended
- In slave mode, a match between the received address and the address enabled in RIICnSER when the value of the received R/W# bit is 0 (including cases where the received address is the general call address)
- In slave transmit mode, a restart condition is detected (a restart condition is detected with RIICnCR2.BBSY = 1 and RIICnCR2.MST = 0)
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

MST Bit (Master/Slave Mode)

This bit indicates master or slave mode.

The RIIC is in slave mode when the MST bit is set to 0 and is in master mode when the bit is set to 1. Combination of this bit and the TRS bit indicates the operating mode of the RIIC.

The value of the MST bit is automatically changed to the value for master mode or slave mode by detection or issuing of a start condition, etc.

[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)

[Clearing conditions]

- When a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

BBSY Flag (Bus Busy Detection)

The BBSY flag indicates whether the I^{2}C bus is occupied (bus busy) or released (bus free).

This bit is set to 1 when the SDA line changes from high to low under the condition of SCL = high, assuming that a start condition has been issued.

When the SDA line changes from low to high under the condition of SCL = high, this bit is cleared to 0 after the bus free time (specified in RIICnBRL) start condition is not detected, assuming that a stop condition has been issued.

[Setting condition]

- When a start condition is detected

[Clearing conditions]

- When the bus free time (specified in RIICnBRL) start condition is not detected after detecting a stop condition
- When 1 is written to the RIICnCR1.IICRST bit with the RIICnCR1.ICE bit set to 0 (RIIC reset)

26.3.3 RIICnMR1 — I^2C Bus Mode Register 1

Access Size: RIICnMR1 is a 32-bit readable/writable register.
RIICnMR1L and RIICnMR1H are 16-bit readable/writable registers.
RIICnMR1LL, RIICnMR1LH, RIICnMR1HL, and RIICnMR1HH are 8-bit readable/writable registers.

Address(es): RIICnMR1: <RIICn_base> + H’0008
RIICnMR1L: <RIICn_base> + H’0008, RIICnMR1H: <RIICn_base> + H’000A
RIICnMR1LL: <RIICn_base> + H’0008, RIICnMR1LH: <RIICn_base> + H’0009,
RIICnMR1HL: <RIICn_base> + H’000A, RIICnMR1HH: <RIICn_base> + H’000B

Initial Value: H’0000 0008 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CKS[2:0]			BCWP	BC[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
6 to 4	CKS[2:0]	All 0	R/W	Internal Reference Clock (IICϕ) Selection b6 b4 0 0 0: IICϕ = P0ϕ/1 0 0 1: IICϕ = P0ϕ/2 0 1 0: IICϕ = P0ϕ/4 0 1 1: IICϕ = P0ϕ/8 1 0 0: IICϕ = P0ϕ/16 1 0 1: IICϕ = P0ϕ/32 1 1 0: IICϕ = P0ϕ/64 1 1 1: IICϕ = P0ϕ/128
3	BCWP*[1]	1	R/W	BC Write Protect 0: Enables a value to be written in the BC[2:0] bits. (This bit is read as 1.)
2 to 0	BC[2:0]	All 0	R/W	Bit Counter b2 b0 0 0 0: 9 bits 0 0 1: 2 bits 0 1 0: 3 bits 0 1 1: 4 bits 1 0 0: 5 bits 1 0 1: 6 bits 1 1 0: 7 bits 1 1 1: 8 bits

Note 1. When rewriting the BC[2:0] bits, write 0 to the BCWP bit simultaneously.

BC[2:0] Bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a rising edge on the SCL line. Although these bits are writable and readable, it is not necessary to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one (data is transferred with an additional acknowledge bit) between transferred frames when the SCL line is at a low level.

The values of the BC[2:0] bits return to 000b at the end of a data transfer including the acknowledge bit or when a start condition including a restart condition is detected.

26.3.4 RIICnMR2 — I^{2}C Bus Mode Register 2

Access Size: RIICnMR2 is a 32-bit readable/writable register.
RIICnMR2L and RIICnMR2H are 16-bit readable/writable registers.
RIICnMR2LL, RIICnMR2LH, RIICnMR2HL, and RIICnMR2HH are 8-bit readable/writable registers.

Address(es): RIICnMR2: <RIICn_base> + H’000C
RIICnMR2L: <RIICn_base> + H’000C, RIICnMR2H: <RIICn_base> + H’000E
RIICnMR2LL: <RIICn_base> + H’000C, RIICnMR2LH: <RIICn_base> + H’000D,
RIICnMR2HL: <RIICn_base> + H’000E, RIICnMR2HH: <RIICn_base> + H’000F

Initial Value: H’0000 0006 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DLCS	SDDL[2:0]			—	TMOH	TMOL	TMOS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	DLCS	0	R/W	SDA Output Delay Clock Source Selection 0: The internal reference clock (IICϕ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (IICϕ/2) is selected as the clock source of the SDA output delay counter.*1
6 to 4	SDDL[2:0]	All 0	R/W	SDA Output Delay Counter • When RIICnMR2.DLCS = 0 (IICϕ) b6 b4 0 0 0: No output delay 0 0 1: 1 IICϕ cycle 0 1 0: 2 IICϕ cycles 0 1 1: 3 IICϕ cycles 1 0 0: 4 IICϕ cycles 1 0 1: 5 IICϕ cycles 1 1 0: 6 IICϕ cycles 1 1 1: 7 IICϕ cycles • When RIICnMR2.DLCS = 1 (IICϕ/2) b6 b4 0 0 0: No output delay 0 0 1: 1 or 2 IICϕ cycles 0 1 0: 3 or 4 IICϕ cycles 0 1 1: 5 or 6 IICϕ cycles 1 0 0: 7 or 8 IICϕ cycles 1 0 1: 9 or 10 IICϕ cycles 1 1 0: 11 or 12 IICϕ cycles 1 1 1: 13 or 14 IICϕ cycles
3	—	0	R	Reserved These bits are read as 0. The write value should be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	TMOH	1	R/W	Timeout H Count Control 0: Count is disabled while the SCL line is at a high level. 1: Count is enabled while the SCL line is at a high level.
1	TMOL	1	R/W	Timeout L Count Control 0: Count is disabled while the SCL line is at a low level. 1: Count is enabled while the SCL line is at a low level.
0	TMOS	0	R/W	Timeout Detection Time Selection 0: Long mode is selected. 1: Short mode is selected.

Note 1. The setting DLCS = 1 (IICϕ/2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting DLCS = 1 becomes invalid and the clock source becomes the internal reference clock (IICϕ).

TMOS Bit (Timeout Detection Time Selection)

This bit is used to select long mode or short mode for the timeout detection time when the timeout function is enabled (RIICnFER.TMOE bit = 1). When this bit is set to 0, long mode is selected. When this bit is set to 1, short mode is selected. In long mode, the timeout detection internal counter functions as a 16 bit-counter. In short mode, the counter functions as a 14 bit-counter. While the SCL line is in the state that enables this counter as specified by bits TMOH and TMOL, the counter counts up in synchronization with the internal reference clock (IICϕ) as a count source.

For details on the timeout function, see **Section 26.13.1, Timeout Function**.

TMOL Bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held low when the timeout function is enabled (RIICnFER.TMOE bit = 1).

TMOH Bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held high when the timeout function is enabled (RIICnFER.TMOE bit = 1).

SDDL[2:0] Bits (SDA Output Delay Setup Counter)

The SDA output can be delayed by the SDDL[2:0] setting. This counter works with the clock source selected by the DLCS bit. The setting of this function can be used for all types of SDA output, including the transmission of the acknowledge bit.

For details on this function, see **Section 26.7, Facility for Delaying SDA Output**.

CAUTION

Set the SDA output delay time to meet the I^{2}C bus standard (within the data enable time/acknowledge enable time*[1]) or the SMBus standard (within the data hold time: 300 [ns] or more, and SCL-clock low-level period - the data setup time: 250 [ns]). Note that, if a value outside the standard is set, communication with communication devices may malfunction or it may seemingly become a start condition or stop condition depending on the bus state.

Note 1. Data enable time/acknowledge enable time
3,450 [ns] (0 to 100 [kbps]: standard mode (Sm))
900 [ns] (0 to 400 [kbps]: fast mode (Fm))
450 [ns] (0 to 1 [Mbps]: fast mode plus (Fm+))

26.3.5 RIICnMR3 — I^{2}C Bus Mode Register 3

Access Size:	RIICnMR3 is a 32-bit readable/writable register. RIICnMR3L and RIICnMR3H are 16-bit readable/writable registers. RIICnMR3LL, RIICnMR3LH, RIICnMR3HL, and RIICnMR3HH are 8-bit readable/writable registers.
Address(es):	RIICnMR3: <RIICn_base> + H’0010 RIICnMR3L: <RIICn_base> + H’0010, RIICnMR3H: <RIICn_base> + H’0012 RIICnMR3LL: <RIICn_base> + H’0010, RIICnMR3LH: <RIICn_base> + H’0011, RIICnMR3HL: <RIICn_base> + H’0012, RIICnMR3HH: <RIICn_base> + H’0013
Initial Value:	H’0000 0000 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SMBE	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	SMBE	0	R/W	SMBus/I^{2}C Bus Selection 0: I^{2}C bus is selected. 1: SMBus is selected.
6	WAIT*[2]	0	R/W	WAIT 0: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low-hold is released by reading RIICnDRR.
5	RDRFS*[2]	0	R/W	RDRF Flag Set Timing Selection 0: The RDRF flag is set at the rising edge of the ninth SCL clock cycle. (The SCL line is not held low at the falling edge of the eighth clock cycle.) 1: The RDRF flag is set at the rising edge of the eighth SCL clock cycle. (The SCL line is held low at the falling edge of the eighth clock cycle.) Low-hold is released by writing a value to the ACKBT bit.
4	ACKWP*[1]	0	W	ACKBT Write Protect 0: Modification of the ACKBT bit is disabled. 1: Modification of the ACKBT bit is enabled.
3	ACKBT*[1]	0	R/W	Transmit Acknowledge 0: A 0 is sent as the acknowledge bit (ACK transmission). 1: A 1 is sent as the acknowledge bit (NACK transmission).
2	ACKBR	0	R	Receive Acknowledge 0: A 0 is received as the acknowledge bit (ACK reception). 1: A 1 is received as the acknowledge bit (NACK reception).

Bit	Bit Name	Initial Value	R/W	Description
1, 0	NF[1:0]	All 0	R/W	Noise Filter Stage Selection b1 b0 0 0: Noise of up to one IICϕ cycle is filtered out (single-stage filter). 0 1: Noise of up to two IICϕ cycles is filtered out (2-stage filter). 1 0: Noise of up to three IICϕ cycles is filtered out (3-stage filter). 1 1: Noise of up to four IICϕ cycles is filtered out (4-stage filter).

Note 1. If it is attempted to write 1 to both ACKWP and ACKBT bits, the ACKBT bit cannot be set to 1.

Note 2. The WAIT and RDRFS bits are valid only in receive mode (invalid in transmit mode).

NF[1:0] Bits (Noise Filter Stage Selection)

These bits are used to select the width of noise that can be removed from the signals input to RIICnSCL or RIICnSDA pin.

ACKBR Bit (Receive Acknowledge)

This bit is used to store the acknowledge bit information received from the receive device in transmit mode.

[Setting condition]

When 1 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1

[Clearing conditions]

- When 0 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

ACKBT Bit (Transmit Acknowledge)

This bit is used to set the bit to be sent at the acknowledge timing in receive mode.

[Setting condition]

When 1 is written to this bit with the ACKWP bit set to 1

[Clearing conditions]

- When 0 is written to this bit with the ACKWP bit set to 1
- When stop condition issuance is detected
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

CAUTION

The ACKBT bit must be written to while the ACKWP bit is 1. If the ACKBT bit is written to with the ACKWP bit cleared to 0, writing to the ACKBT bit is disabled.

ACKWP Bit (ACKBT Write Protect)

This bit is used to control the modification of the ACKBT bit.

RDRFS Bit (RDRF Flag Set Timing Selection)

This bit is used to select the RDRF flag set timing in receive mode and also to select whether to hold the SCL line low at the falling edge of the eighth SCL clock cycle.

When the RDRFS bit is 0, the SCL line is not held low at the falling edge of the eighth SCL clock cycle, and the RDRF flag is set to 1 at the rising edge of the ninth SCL clock cycle.

When the RDRFS bit is 1, the RDRF flag is set to 1 at the rising edge of the eighth SCL clock cycle and the SCL line is held low at the falling edge of the eighth SCL clock cycle. The low-hold of the SCL line is released by writing a value to the ACKBT bit.

After data is received with this setting, the SCL line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKBT = 0) or NACK (ACKBT = 1) according to receive data.

WAIT Bit (WAIT)

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (RIICnDRR) is completely read each time single-byte data is received in receive mode. When the WAIT bit is 0, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the RDRFS and WAIT bits are 0, continuous receive operation is enabled with the double buffer.

When the WAIT bit is 1, the SCL line is held low from the falling edge of the ninth clock cycle until the RIICnDRR value is read each time single-byte data is received. This enables receive operation in byte units.

CAUTION

When the value of the WAIT bit is to be read, be sure to read the RIICnDRR beforehand.

SMBE Bit (SMBus Select)

Setting this bit to 1 enables the RIICnSER.HOAE bit.

26.3.6 RIICnFER — I^{2}C Bus Function Enable Register

Access Size: RIICnFER is a 32-bit readable/writable register.
RIICnFERL and RIICnFERH are 16-bit readable/writable registers.
RIICnFERLL, RIICnFERLH, RIICnFERHL, and RIICnFERHH are 8-bit readable/writable registers.

Address(es): RIICnFER: <RIICn_base> + H’0014
RIICnFERL: <RIICn_base> + H’0014, RIICnFERH: <RIICn_base> + H’0016
RIICnFERLL: <RIICn_base> + H’0014, RIICnFERLH: <RIICn_base> + H’0015,
RIICnFERHL: <RIICn_base> + H’0016, RIICnFERHH: <RIICn_base> + H’0017

Initial Value: H’0000 0072 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	FMPE	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	FMPE	0	R/W	Fast-mode Plus Enable 0: No fm+ slope control circuit is used for the SCLn pin and SDAn pin 1: An fm+ slope control circuit is used for the SCLn pin and SDAn pin
6	SCLE	1	R/W	SCL Synchronous Circuit Enable 0: No SCL synchronous circuit is used. 1: An SCL synchronous circuit is used.
5	NFE	1	R/W	Digital Noise Filter Circuit Enable 0: No digital noise filter circuit is used. 1: A digital noise filter circuit is used.
4	NACKE	1	R/W	NACK Reception Transfer Suspension Enable 0: Transfer operation is not suspended during NACK reception (transfer suspension disabled). 1: Transfer operation is suspended during NACK reception (transfer suspension enabled).
3	SALE	0	R/W	Slave Arbitration-Lost Detection Enable 0: Slave arbitration-lost detection is disabled. 1: Slave arbitration-lost detection is enabled.
2	NALE	0	R/W	NACK Transmission Arbitration-Lost Detection Enable 0: NACK transmission arbitration-lost detection is disabled. 1: NACK transmission arbitration-lost detection is enabled.
1	MALE	1	R/W	Master Arbitration-Lost Detection Enable 0: Master arbitration-lost detection is disabled. (Disables the arbitration-lost detection function and does not clear the RIICnCR2.MST and TRS bits automatically when arbitration is lost.) 1: Master arbitration-lost detection is enabled. (Enables the arbitration-lost detection function and clears the RIICnCR2.MST and TRS bits automatically when arbitration is lost.)

Bit	Bit Name	Initial Value	R/W	Description
0	TMOE	0	R/W	Timeout Function Enable 0: The timeout function is disabled. 1: The timeout function is enabled.

TMOE Bit (Timeout Function Enable)

This bit is used to enable or disable the timeout function.

For details on the timeout function, see **Section 26.13.1, Timeout Function**.

MALE Bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1.

NALE Bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in receive mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).

SALE Bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmit mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).

NACKE Bit (NACK Reception Transfer Suspension Enable)

This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received in transmit mode. Normally, set this bit to 1.

When NACK is received with the NACKE bit set to 1, the next transfer operation is suspended.

When the NACKE bit is 0, the next transfer operation is continued regardless of the received acknowledge content.

SCLE Bit (SCL Synchronous Circuit Enable)

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1. When the SCLE bit is cleared to 0 (SCL synchronous circuit not used), the RIIC does not synchronize the SCL clock with the SCL input clock. In this setting, the RIIC outputs the SCL clock with the transfer rate set in RIICnBRH and RIICnBRL regardless of the SCL line state. For this reason, if the bus load of the I^{2}C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output. When no SCL synchronous circuit is used, it also affects the issuance of a start condition, restart condition, and stop condition, and the continuous output of extra SCL clock cycles.

This bit must not be cleared to 0 except for checking the output of the transfer rate.

FMPE Bit (Fast-mode Plus Enable)

This bit is used to specify whether to use a slope control circuit for Fast-mode Plus [fm+].

When this bit is set to 1, a slope control circuit conforming to the Fast-mode Plus [fm+] slope control standard (tof) of the I^{2}C bus is selected. When this bit is cleared to 0, a slope control circuit conforming to the Standard-mode [Sm] and Fast- mode [fm] slope control standard (tof) of the I^{2}C bus is selected.

Set this bit to 1 when using the transmission rate within a range up to 1 Mbps (Fast-mode Plus[fm+]) of the I^{2}C bus standard. Clear this bit to 0 when using the transmission rate at other rates (up to 100 kbps[Sm], up to 400 kbps[fm]) or for SMBus (10 to 100 kbps).

26.3.7 RIICnSER — I^{2}C Bus Status Enable Register

Access Size:	RIICnSER is a 32-bit readable/writable register. RIICnSERL and RIICnSERH are 16-bit readable/writable registers. RIICnSERLL, RIICnSERLH, RRIICnSERHL, and RIICnSERHH are 8-bit readable/writable registers.
Address(es):	RIICnSER: <RIICn_base> + H'0018 RIICnSERL: <RIICn_base> + H'0018, RIICnSERH: <RIICn_base> + H'001A RIICnSERLL: <RIICn_base> + H'0018, RIICnSERLH: <RIICn_base> + H'0019, RIICnSERHL: <RIICn_base> + H'001A, RIICnSERHH: <RIICn_base> + H'001B
Initial Value:	H'0000 0009 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HOAE	—	DIDE	—	GCE	SAR2	SAR1	SAR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	HOAE	0	R/W	Host Address Enable 0: Host address detection is disabled. 1: Host address detection is enabled.
6	—	0	R	Reserved This bit is read as 0. The write value should be 0.
5	DIDE	0	R/W	Device-ID Address Detection Enable 0: Device-ID address detection is disabled. 1: Device-ID address detection is enabled.
4	—	0	R	Reserved This bit is read as 0. The write value should be 0.
3	GCE	1	R/W	General Call Address Enable 0: General call address detection is disabled. 1: General call address detection is enabled.
2	SAR2	0	R/W	Slave Address Register 2 Enable 0: Slave address in RIICnSAR2 is disabled. 1: Slave address in RIICnSAR2 is enabled.
1	SAR1	0	R/W	Slave Address Register 1 Enable 0: Slave address in RIICnSAR1 is disabled. 1: Slave address in RIICnSAR1 is enabled.
0	SAR0	1	R/W	Slave Address Register 0 Enable 0: Slave address in RIICnSAR0 is disabled. 1: Slave address in RIICnSAR0 is enabled.

SARy Bit (Slave Address Register y Enable) (y = 0 to 2)

This bit is used to enable or disable the received slave address and the slave address set in RIICnSARy.

When this bit is set to 1, the slave address set in RIICnSARy is enabled and is compared with the received slave address. When this bit is cleared to 0, the slave address set in RIICnSARy is disabled and is ignored even if it matches the received slave address.

GCE Bit (General Call Address Enable)

This bit is used to specify whether to ignore the general call address (0000 000b + 0 [W]: All 0) when it is received. When this bit is set to 1, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in RIICnSARy (y = 0 to 2) and performs data receive operation.

When this bit is cleared to 0, the received slave address is ignored even if it matches the general call address.

DIDE Bit (Device-ID Address Detection Enable)

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID (1111 100b) is received in the first frame after a start condition or restart condition is detected.

When this bit is set to 1, if the received first frame matches the device ID, the RIIC recognizes that the Device-ID address has been received. When the following R/W# bit is 0 [W], the RIIC recognizes the second and the following frames as slave addresses and continues the receive operation.

When this bit is cleared to 0, the RIIC ignores the received first frame even if it matches the device ID address and recognizes the first frame as a normal slave address.

For details on the device-ID address detection, see **Section 26.9.3, Device-ID Address Detection**.

HOAE Bit (Host Address Enable)

This bit is used to specify whether to ignore received host address (0001 000b) when the RIICnMR3.SMBS bit is 1. When this bit is set to 1 while the RIICnMR3.SMBS bit is 1, if the received slave address matches the host address, the RIIC recognizes the received slave address as the host address independently of the slave addresses set in RIICnSARy (y = 0 to 2) and performs the receive operation.

When the RIICnMR3.SMBS bit or the HOAE bit is cleared to 0, the received slave address is ignored even if it matches the host address.

26.3.8 RIICnIER — I^{2}C Bus Interrupt Enable Register

Access Size:	RIICnIER is a 32-bit readable/writable register. RIICnIERL and RIICnIERH are 16-bit readable/writable registers. RIICnIERLL, RIICnIERLH, RIICnIERHL, and RIICnIERHH are 8-bit readable/writable registers.
Address(es):	RIICnIER: <RIICn_base> + H’001C RIICnIERL: <RIICn_base> + H’001C, RIICnIERH: <RIICn_base> + H’001E RIICnIERLL: <RIICn_base> + H’001C, RIICnIERLH: <RIICn_base> + H’001D, RIICnIERHL: <RIICn_base> + H’001E, RIICnIERHH: <RIICn_base> + H’001F
Initial Value:	H’0000 0000 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	TIE	0	R/W	Transmit Data Empty Interrupt Enable 0: Transmit data empty interrupt request (INTRIICTI) is disabled. 1: Transmit data empty interrupt request (INTRIICTI) is enabled.
6	TEIE	0	R/W	Transmit End Interrupt Enable 0: Transmit end interrupt request (INTRIICTEI) is disabled. 1: Transmit end interrupt request (INTRIICTEI) is enabled.
5	RIE	0	R/W	Receive Data Full Interrupt Enable 0: Receive data full interrupt request (INTRIICRI) is disabled. 1: Receive data full interrupt request (INTRIICRI) is enabled.
4	NAKIE	0	R/W	NACK Reception Interrupt Enable 0: NACK reception interrupt request (INTRIICNAKI) is disabled. 1: NACK reception interrupt request (INTRIICNAKI) is enabled.
3	SPIE	0	R/W	Stop Condition Detection Interrupt Enable 0: Stop condition detection interrupt request (INTRIICSPI) is disabled. 1: Stop condition detection interrupt request (INTRIICSPI) is enabled.
2	STIE	0	R/W	Start Condition Detection Interrupt Enable 0: Start condition detection interrupt request (INTRIICSTI) is disabled. 1: Start condition detection interrupt request (INTRIICSTI) is enabled.
1	ALIE	0	R/W	Arbitration-Lost Interrupt Enable 0: Arbitration-lost interrupt request (INTRIICALI) is disabled. 1: Arbitration-lost interrupt request (INTRIICALI) is enabled.
0	TMOIE	0	R/W	Timeout Interrupt Enable 0: Timeout interrupt request (INTRIICTMOI) is disabled. 1: Timeout interrupt request (INTRIICTMOI) is enabled.

TMOIE Bit (Timeout Interrupt Enable)

This bit is used to enable or disable timeout interrupt requests (INTRIICTMOI) when the RIICnSR2.TMOF flag is set to 1. An INTRIICTMOI interrupt request is canceled by clearing the TMOF flag or the TMOIE bit to 0.

ALIE Bit (Arbitration-Lost Interrupt Enable)

This bit is used to enable or disable arbitration-lost interrupt requests (INTRIICALII) when the RIICnSR2.AL flag is set to 1. An INTRIICALII interrupt request is canceled by clearing the AL flag or the ALIE bit to 0.

STIE Bit (Start Condition Detection Interrupt Enable)

This bit is used to enable or disable start condition detection interrupt requests (INTRIICSTI) when the RIICnSR2.START flag is set to 1. An INTRIICSTI interrupt request is canceled by clearing the START flag or the STIE bit to 0.

SPIE Bit (Stop Condition Detection Interrupt Enable)

This bit is used to enable or disable stop condition detection interrupt requests (INTRIICSPI) when the RIICnSR2.STOP flag is set to 1. An INTRIICSPI interrupt request is canceled by clearing the STOP flag or the SPIE bit to 0.

NAKIE Bit (NACK Reception Interrupt Enable)

This bit is used to enable or disable NACK reception interrupt requests (INTRIICNAKI) when the RIICnSR2.NACKF flag is set to 1. An INTRIICNAKI interrupt request is canceled by clearing the NACKF flag or the NAKIE bit to 0.

RIE Bit (Receive Data Full Interrupt Enable)

This bit is used to enable or disable receive data full interrupt requests (INTRIICRI) when the RIICnSR2.RDRF flag in ICSR2 is set to 1. An INTRIICRI interrupt request is canceled by clearing the RDRF flag or the RIE bit to 0.

TEIE Bit (Transmit End Interrupt Enable)

This bit is used to enable or disable transmit end interrupts (INTRIICTEI) when the RIICnSR2.TEND flag is set to 1. An INTRIICTEI interrupt request is canceled by clearing the TEND flag or the TEIE bit to 0.

TIE Bit (Transmit Data Empty Interrupt Enable)

This bit is used to enable or disable transmit data empty interrupts (INTRIICTI) when the RIICnSR2.TDRE flag is set to 1.

26.3.9 RIICnSR1 — I^{2}C Bus Status Register 1

Access Size: RIICnSR1 is a 32-bit readable/writable register.
RIICnSR1L and RIICnSR1H are 16-bit readable/writable registers.
RIICnSR1LL, RIICnSR1LH, RIICnSR1HL, and RIICnSR1HH are 8/1-bit readable/writable registers.

Address(es): RIICnSR1: <RIICn_base> + H’0020
RIICnSR1L: <RIICn_base> + H’0020, RIICnSR1H: <RIICn_base> + H’0022
RIICnSR1LL: <RIICn_base> + H’0020, RIICnSR1LH: <RIICn_base> + H’0021,
RIICnSR1HL: <RIICn_base> + H’0022, RIICnSR1HH: <RIICn_base> + H’0023

Initial Value: H’0000 0000 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HOA	—	DID	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R(/W)*1	R	R(/W)*1	R	R/W*1	R(/W)*1	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	HOA	0	R(/W)*1	Host Address Detection Flag 0: Host address is not detected. 1: Host address is detected.
6	—	0	R	Reserved This bit is read as 0. The write value should be 0.
5	DID	0	R(/W)*1	Device-ID Address Detection Flag 0: Device-ID command is not detected. 1: Device-ID command is detected.
4	—	0	R	Reserved This bit is read as 0. The write value should be 0.
3	GCA	0	R/W*1	General Call Address Detection Flag 0: General call address is not detected. 1: General call address is detected.
2	AAS2	0	R(/W)*1	Slave Address 2 Detection Flag 0: Slave address 2 is not detected. 1: Slave address 2 is detected.
1	AAS1	0	R(/W)*1	Slave Address 1 Detection Flag 0: Slave address 1 is not detected. 1: Slave address 1 is detected.
0	AAS0	0	R(/W)*1	Slave Address 0 Detection Flag 0: Slave address 0 is not detected. 1: Slave address 0 is detected.

Note 1. Only 0 can be written to this bit.

AASy Flag (Slave Address y Detection) (y = 0 to 2)

[Setting conditions]

<For 7-bit address format: RIICnSARy.FSy = 0>

When the received slave address matches the RIICnSARy.SVA[7:1] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

<For 10-bit address format: RIICnSARy.FSy = 1>

When the received slave address matches a value of (1111 0b + RIICnSARy.SVA[9:8]) and the following address matches the RIICnSARy.SVA[7:0] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)
This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the AASy bit after reading AASy = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

<For 7-bit address format: RIICnSARy.FSy = 0>

- When the received slave address does not match the RIICnSARy.SVA[7:1] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

<For 10-bit address format: RIICnSARy.FSy = 1>

- When the received slave address does not match a value of (1111 0b + RIICnSARy.SVA[9:8]) with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the received slave address matches a value of (1111 0b + RIICnSARy.SVA[9:8]) and the following address does not match the RIICnSARy.SVA[7:0] value with the RIICnSER.SARy bit set to 1 (slave address y detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

GCA Flag (General Call Address Detection)

[Setting condition]

When the received slave address matches the general call address (0000 000b + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the GCA bit after reading GCA = 1
- When a stop condition is detected
- When the received slave address does not match the general call address (0000 000b + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

DID Flag (Device-ID Address Detection)

[Setting condition]

- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100b) + 0 [W]) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled) This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.
- When a restart condition is detected after a match with the device ID address and the device ID address (1111 100b) + 1 [R] has matched with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the DID bit after reading DID = 1
- When a stop condition is detected
- When the first frame received immediately after a start condition or restart condition is detected does not match a value of (device ID (1111 100b)) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled) This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100b) + 0 [W]) and the second frame does not match any of slave addresses 0 to 2 with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

HOA Flag (Host Address Detection)

[Setting condition]

When the received slave address matches the host address (0001 000b) while the RIICnMR3.SMBE bit and RIICnSER.HOAE bit are set to 1 (host address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the HOA bit after reading HOA = 1
- When a stop condition is detected
- When 0 is written to the RIICnMR3.SMBS bit in ICMR3 or the RIICnSER.HOAE bit
- When the received slave address does not match the host address (0001 000b) with the RIICnSER.HOAE bit set to 1 (host address detection enabled)
  This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit 1 to apply an RIIC reset or an internal reset

26.3.10 RIICnSR2 — I^{2}C Bus Status Register 2

Access Size:	RIICnSR2 is a 32-bit readable/writable register. RIICnSR2L and RIICnSR2H are 16-bit readable/writable registers. RIICnSR2LL, RIICnSR2LH, RIICnSR2HL, and RIICnSR2HH are 8/1-bit readable/writable registers.
Address(es):	RIICnSR2: <RIICn_base> + H'0024 RIICnSR2L: <RIICn_base> + H'0024, RIICnSR2H: <RIICn_base> + H'0026 RIICnSR2LL: <RIICn_base> + H'0024, RIICnSR2LH: <RIICn_base> + H'0025, RIICnSR2HL: <RIICn_base> + H'0026, RIICnSR2HH: <RIICn_base> + H'0027
Initial Value:	H'0000 0000 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TDRE	TEND	RDRF	NACKF	STOP	STAT	AL	TMOF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1	R(/W)*1

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7	TDRE	0	R	Transmit Data Empty Flag 0: RIICnDRT contains transmit data. 1: RIICnDRT contains no transmit data.
6	TEND	0	R(/W)*1	Transmit End Flag 0: Data is being transmitted. 1: Data has been transmitted.
5	RDRF	0	R(/W)*1	Receive Data Full Flag 0: RIICnDRR contains no receive data. 1: RIICnDRR contains receive data.
4	NACKF	0	R(/W)*1	NACK Reception Flag 0: NACK is not received. 1: NACK is received.
3	STOP	0	R(/W)*1	Stop Condition Detection Flag 0: Stop condition is not detected. 1: Stop condition is detected.
2	START	0	R(/W)*1	Start Condition Detection Flag 0: Start condition is not detected. 1: Start condition is detected.
1	AL	0	R(/W)*1	Arbitration-Lost Flag 0: Arbitration is not lost. 1: Arbitration is lost.
0	TMOF	0	R(/W)*1	Timeout Flag 0: No timeout has occurred. 1: Timeout has occurred.

Note 1. Only 0 can be written to this bit.

TMOF Flag (Timeout)

This flag is set to 1 when the RIIC recognizes timeout after the SCL line state remains unchanged for a certain period.
[Setting condition]

The timeout function is enabled when the RIICnFER.TMOE bit is 1. It detects an abnormal bus state that the SCL line is held low or high during the following conditions:

- The bus is busy (RIICnCR2.BBSY = 1) in master mode (RIICnCR2.MST = 1).
- The slave address matches that of this module (RIICnSR1 register is not H’00) and the bus is busy (RIICnCR2.BBSY = 1) in slave mode (RIICnCR2.MST = 0).
- Issuing of a start condition is being requested (RIICnCR2.ST = 1) and the bus is free (RIICnCR2.BBSY = 0).

[Clearing conditions]

- When 0 is written to the TMOF bit after reading TMOF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

AL Flag (Arbitration-Lost)

This flag shows that bus mastership has been lost (loss in arbitration) due to a bus conflict or some other reason when a start condition is issued or an address and data are transmitted. The RIIC monitors the level on the SDA line during transmission and, if the level on the line does not match the value of the bit being output, sets the value of the AL bit to 1 to indicate that the bus is occupied by another device.

The RIIC can also set the flag to indicate the detection of loss of arbitration during NACK transmission in receive mode or during data transmission in slave mode.

[Setting conditions]

<When master arbitration-lost detection is enabled: RIICnFER.MALE = 1>

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmit mode (when the SDA line is driven low while the internal SDA output is at a high level (the SDA pin is in the high-impedance state))
- When a start condition is detected while the RIICnCR2.ST bit is 1 (start condition issuance request) or the internal SDA output state does not match the SDA line level
- When the RIICnCR2.ST bit is set to 1 (start condition issuance request) with the RIICnCR2.BBSY flag set to 1.

<When NACK arbitration-lost detection is enabled: RIICnFER.NALE = 1>

When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock in the ACK period during NACK transmission in receive mode

<When slave arbitration-lost detection is enabled: RIICnFER.SALE = 1>

When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmit mode

[Clearing conditions]

- When 0 is written to the AL bit after reading AL = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

Table 26.7 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions

RIICnFER			RIICnSR2		
MALE	NALE	SALE	AL	Error	Arbitration-Lost Generation Source
1	x	x	1	Start condition issuance error	When internal SDA output state does not match SDA line level when a start condition is detected while the RIICnCR2.ST bit is 1
					When RIICnCR2.ST is set to 1 with RIICnCR2.BBSY set to 1
			1	Transmit data mismatch	When transmit data (including slave address) does not match the bus state in master transmit mode
x	1	x	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master receive mode or slave receive mode
x	x	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmit mode

Note: x: Don’t care

START Flag (Start Condition Detection)

[Setting condition]

When a start condition (or a restart condition) is detected

[Clearing conditions]

- When 0 is written to the START bit after reading START = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

STOP Flag (Stop Condition Detection)

[Setting condition]

When a stop condition is detected

[Clearing conditions]

- When 0 is written to the STOP bit after reading STOP = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

NACKF Flag (NACK Reception)

[Setting condition]

When acknowledge is not received (NACK is received) from the receive device in transmit mode with the RIICnFER.NACKE bit set to 1 (transfer suspension enabled)

[Clearing conditions]

- When 0 is written to the NACKF bit after reading NACKF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

CAUTION

When the NACKF flag is set to 1, the RIIC suspends data transmission/reception. Writing to RIICnDRT in transmit mode or reading from RIICnDRR in receive mode with the NACKF flag set to 1 does not enable data transmit/ receive operation. To restart data transmission/reception, clear the NACKF flag to 0.

RDRF Flag (Receive Data Full)

[Setting conditions]

- Slave receive mode
  - When the received slave address matches and the RIICnCR2.TRS bit is cleared to 0 after a start condition (or a restart condition) is detected
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR
- Master receive mode
  - When the slave address and the data direction are transmitted and the receive mode is entered (the RIICnCR2.TRS bit is set to 1) after a start condition (or a restart condition) is issued
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR

[Clearing conditions]

- When 0 is written to the RDRF bit after reading RDRF = 1
- When data is read from RIICnDRR
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

TEND Flag (Transmit End)

[Setting condition]

At the rising edge of the ninth SCL clock cycle while the TDRE flag is 1

[Clearing conditions]

- When 0 is written to the TEND bit after reading TEND = 1
- When data is written to RIICnDRT
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

TDRE Flag (Transmit Data Empty)

[Setting conditions]

- When data has been transferred from RIICnDRT to RIICnDRS and RIICnDRT becomes empty
- When the RIICnCR2.TRS bit is set to 1
  - When the RIICnCR2.MST bit is set to 1 after a start condition (or a restart condition) is detected
  - When the RIIC enters transmit mode from receive mode
- When the received slave address matches while the TRS bit is 1

[Clearing conditions]

- When data is written to RIICnDRT
- When the RIICnCR2.TRS bit is cleared to 0
  - When a stop condition is detected
  - When the RIIC enters receive mode from transmit mode
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

CAUTION

When the NACKF flag is set to 1 while the RIICnFER.NACKE bit is 1, the RIIC suspends data transmission/ reception. Here, if the TDRE flag is 0 (next transmit data has been written), data is transferred to the RIICnDRS register and the RIICnDRT register becomes empty at the rising edge of the ninth clock cycle, but the TDRE flag is not set to 1.

26.3.11 RIICnSARy — I^{2}C Slave Address Register y (y = 0 to 2)

Access Size: RIICnSARy is a 32-bit readable/writable register.
RIICnSARyL and RIICnSARyH are 16-bit readable/writable registers.
RIICnSARyLL, RIICnSARyLH, RIICnSARyHL, and RIICnSARyHH are 8-bit readable/writable registers.

Address(es): RIICnSAR0: <RIICn_base> + H'0028
RIICnSAR0L: <RIICn_base> + H'0028, RIICnSAR0H: <RIICn_base> + H'002A
RIICnSAR0LL: <RIICn_base> + H'0028, RIICnSAR0LH: <RIICn_base> + H'0029,
RIICnSAR0HL: <RIICn_base> + H'002A, RIICnSAR0HH: <RIICn_base> + H'002B
RIICnSAR1: <RIICn_base> + H'002C
RIICnSAR1L: <RIICn_base> + H'002C, RIICnSAR1H: <RIICn_base> + H'002E
RIICnSAR1LL: <RIICn_base> + H'002C, RIICnSAR1LH: <RIICn_base> + H'002D,
RIICnSAR1HL: <RIICn_base> + H'002E, RIICnSAR1HH: <RIICn_base> + H'002F
RIICnSAR2: <RIICn_base> + H'0030
RIICnSAR2L: <RIICn_base> + H'0030, RIICnSAR2H: <RIICn_base> + H'0032
RIICnSAR2LL: <RIICn_base> + H'0030, RIICnSAR2LH: <RIICn_base> + H'0031,
RIICnSAR2HL: <RIICn_base> + H'0032, RIICnSAR2HH: <RIICn_base> + H'0033

Initial Value: H'0000 0000 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FSy	—	—	—	—	—	SVA[9:1]									SVA0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
15	FSy	0	R/W	7-Bit/10-Bit Address Format Selection 0: The 7-bit address format is selected. 1: The 10-bit address format is selected.
14 to 10	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
9 to 1	SVA[9:1]	All 0	R/W	7-Bit Address/10-Bit Address Upper Bits A slave address is set. ● When the FSy bit is 0 (7-bit address format), the SVA[7:1] bits are Valid and form a 7-bit slave address. ● When the FSy bit is 1 (10-bit address format), SVA[9:1] bits form a 10-bit slave address (combined with the SVA0 bit).
0	SVA0	0	R/W	10-Bit Address LSB The least significant bit (LSB) of a 10-bit slave address is set. ● When the FSy bit is 0 (7-bit address format), this bit is invalid. ● When the FSy bit is 1 (10-bit address format), this bit is a 10-bit slave address (combined with the SVA[9:1] bits).

SVA0 Bit (10-Bit Address LSB)

When the 10-bit address format is selected (RIICnSARy.FSy = 1), this bit functions as the LSB of a 10-bit address and forms a 10-bit address in combination with the SVA[9:1] bits.

When the RIICnSER.SARy bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 1, this bit is valid. While the RIICnSARy.FSy bit or SARy bit is 0, the setting of this bit is ignored.

SVA[9:1] Bits (7-Bit Address/10-Bit Address Upper Bits)

When the 7-bit address format is selected (RIICnSARy.FSy = 0), these bits function as a 7-bit address. When the 10-bit address format is selected (RIICnSARy.FSy = 1), these bits function as a 10-bit address in combination with the SVA0 bit.

While the RIICnSER.SARy bit is 0, the setting of these bits is ignored.

FSy Bit (7-Bit/10-Bit Address Format Selection)

This bit is used to select 7-bit address or 10-bit address for slave address y (in RIICnSARy).

When the RIICnSER.SARy bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 0, the 7-bit address format is selected for slave address y, the RIICnSARy.SVA[7:1] setting is valid, and the settings of the SVA[9:8] bits and the RIICnSARy.SVA0 bit are ignored.

When the RIICnSER.SARy bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 1, the 10-bit address format is selected for slave address y and the settings of the SVA[9:1] bits and the SVA0 bit are valid.

While the RIICnSER.SARy bit is 0 (RIICnSARy disabled), the setting of the RIICnSARy.FSy bit is invalid.

26.3.12 RIICnBRL — I^{2}C Bus Bit Rate Low-Level Register

Access Size:	RIICnBRL is a 32-bit readable/writable register. RIICnBRLL and RIICnBRLH are 16-bit readable/writable registers. RIICnBRLLL, RIICnBRLLH, RIICnBRLHL, and RIICnBRLHH are 8-bit readable/writable registers.
Address(es):	RIICnBRL: <RIICn_base> + H’0034 RIICnBRLL: <RIICn_base> + H’0034, RIICnBRLH: <RIICn_base> + H’0036 RIICnBRLLL: <RIICn_base> + H’0034, RIICnBRLLH: <RIICn_base> + H’0035, RIICnBRLHL: <RIICn_base> + H’0036, RIICnBRLHH: <RIICn_base> + H’0037
Initial Value:	H’0000 00FF This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BRL[4:0]				
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7 to 5	—	All 1	R	Reserved These bits are read as 1. The write value should be 1.
4 to 0	BRL[4:0]	All 1	R/W	Bit Rate Low-Level Period Low-level period of SCL clock

The RIICnBRL register is a 5-bit register that is used to set the width at low level for the SCL clock.

It also works to generate the data setup time for automatic SCL low-hold operation (see **Section 26.10, Automatically Low-Hold Function for SCL**); when the RIIC is used only in slave mode, this register needs to be set to a value at least the same as the data setup time*[1].

RIICnBRL counts the low-level period with the internal reference clock source (IICϕ) specified by the RIICnMR1.CKS[2:0] bits.

Note 1. Data setup time (tSU: DAT)

250 [ns] (0 to 100 [kbps]: standard mode (Sm))

100 [ns] (0 to 400 [kbps]: fast mode (Fm))

50 [ns] (0 to 1 [Mbps]: fast mode plus (Fm+))

26.3.13 RIICnBRH — I^{2}C Bus Bit Rate High-Level Register

Access Size: RIICnBRH is a 32-bit readable/writable register.
RIICnBRHL and RIICnBRHH are 16-bit readable/writable registers.
RIICnBRHLL, RIICnBRHLH, RIICnBRHHL, and RIICnBRHHH are 8-bit readable/writable registers.

Address(es): RIICnBRH: <RIICn_base> + H’0038
RIICnBRHL: <RIICn_base> + H’0038, RIICnBRHH: <RIICn_base> + H’003A
RIICnBRHLL: <RIICn_base> + H’0038, RIICnBRHLH: <RIICn_base> + H’0039,
RIICnBRHHL: <RIICn_base> + H’003A, RIICn- BRHHH:<RIICn_base> + H’003B

Initial Value: H’0000 00FF This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BRH[4:0]				
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as 0. The write value should be 0.
7 to 5	—	All 1	R	Reserved These bits are read as 1. The write value should be 1.
4 to 0	BRH[4:0]	All 1	R/W	Bit Rate High-Level Period High-level period of SCL clock

RIICnBRH is a 5-bit register to set the high-level period of SCL clock. RIICnBRH is valid in master mode. If the RIIC is used only in slave mode, this register need not to set the high-level period.

RIICnBRH counts the high-level period with the internal reference clock source ($IIC\phi$) specified by the RIICnMR1.CKS[2:0] bits in ICMR1.

The frequency and duty cycle are calculated using one of the following expressions (1) to (5) according to the register settings.

CAUTION

The minimum value that can be specified in RIICnBRL and RIICnBRH is determined according to the values of the SCLE and NFE bits in RIICnFER and the NF bit in RIICnMR3. For details of the minimum specifiable value, see **Table 26.8**.

(1) When SCLE = 0
$\text{Frequency} = 1 / \{[(BRH + 1) + (BRL + 1)] / IIC\phi + tr + tf\}$
$\text{Duty cycle} = \{tr + (BRH + 1) / IIC\phi\} / \{tr + tf + [(BRH + 1) + (BRL + 1)] / IIC\phi\}$

(2) When SCLE = 1, NFE = 0, CKS = 000 ($IIC\phi = P0\phi$) $\text{Frequency} = 1 / \{[(BRH + 3) + (BRL + 3)] / IIC\phi + tr + tf\}$
$\text{Duty cycle} = \{tr + (BRH + 3) / IIC\phi\} / \{tr + tf + [(BRH + 3) + (BRL + 3)] / IIC\phi\}$

(3) When SCLE = 1, NFE = 1, CKS = 000 ($IIC\phi = P0\phi$)
Frequency = $1 / \{[(BRH + 3 + nf) + (BRL + 3 + nf)] / IIC\phi + tr + tf\}$
Duty cycle = $\{tr + (BRH + 3 + nf) / IIC\phi\} / \{tr + tf + [(BRH + 3 + nf) + (BRL + 3 + nf)] / IIC\phi\}$

(4) When SCLE = 1, NFE = 0, CKS ^{1} 000 ($IIC\phi < P0\phi$)
Frequency = $1 / \{[(BRH + 2) + (BRL + 2)] / IIC\phi + tr + tf\}$
Duty cycle = $\{tr + (BRH + 2) / IIC\phi\} / \{tr + tf + [(BRH + 2) + (BRL + 2)] / IIC\phi\}$

(5) When SCLE = 1, NFE = 1, CKS ^{1} 000 ($IIC\phi < P0\phi$)
Frequency = $1 / \{[(BRH + 2 + nf) + (BRL + 2 + nf)] / IIC\phi + tr + tf\}$
Duty cycle = $\{tr + (BRH + 2 + nf) / IIC\phi\} / \{tr + tf + [(BRH + 2 + nf) + (BRL + 2 + nf)] / IIC\phi\}$

Symbols in the expressions
SCLE: RIICnFER.SCLE bit
BRH: RIICnBRH.BRH[4:0] bits
BRL: RIICnBRL.BRL[4:0] bits
CKS: RIICnMR1.CKS bits
NFE: RIICnFER.NFE bit
IICϕ: Internal reference clock selected by the CKS bits
tf: SCL signal falling time [s]*[1]
tr: SCL signal rising time [s]* [1]
nf: Number of digital noise filter stages specified in the RIICnMR3.NF[1:0] bits

Note 1. The rising time (tr) and falling time (tf) of the SCL signal depend on the total capacitance of the bus line (Cb) and pull-up resistor (Rp). For details, see I²C Bus Standard from NXP Semiconductors.

Table 26.8 Minimum Specifiable Value for RIICnBRL and RIICnBRH

SCLE	NFE	nf	Minimum Pulse Width that Passes through Digital Filter	Minimum Specifiable Value for BRH and BRL	Pulse Width when Minimum Value is Specified
0	0	—	1 × IICϕ	1	2 × IICϕ
0	1	1	2 × IICϕ	2	3 × IICϕ
0	1	2	3 × IICϕ	3	4 × IICϕ
0	1	3	4 × IICϕ	4	5 × IICϕ
0	1	4	5 × IICϕ	5	6 × IICϕ
IICϕ cycle > P0ϕ cycle (CKS ≠ 000)					
1	0	—	1 × IICϕ	0	2 × IICϕ
1	1	1	2 × IICϕ	1	4 × IICϕ
1	1	2	3 × IICϕ	2	6 × IICϕ
1	1	3	4 × IICϕ	3	8 × IICϕ
1	1	4	5 × IICϕ	4	10 × IICϕ
IICϕ cycle = P0ϕ cycle (CKS = 000)					
1	0	—	2 × P0ϕ	0	3 × IICϕ
1	1	1	3 × P0ϕ	1	5 × IICϕ
1	1	2	4 × P0ϕ	2	7 × IICϕ
1	1	3	5 × P0ϕ	3	9 × IICϕ
1	1	4	6 × P0ϕ	4	11 × IICϕ

Table 26.9 and **Table 26.10** list examples of RIICnBRH/RIICnBRL settings.

Table 26.9 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1 and RIICnFER.NFE= 0)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P0ϕ [MHz]			
	100			
	RIICnMR1.CKS[2:0]	RIICnBRH.BRH	RIICnBRL.BRL	Actual transfer rate [kbps]
12	111b	30 (H’FE)	31 (H’FF)	12
50	101b	28 (H’FC)	30 (H’FE)	50.4
100	100b	28 (H’FC)	31 (H’FF)	99.2
400	010b	29 (H’FD)	30 (H’FE)	396.8
1000	001b	20 (H’F4)	25 (H’F9)	1020.4

Table 26.10 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1, RIICnFER.NFE= 1, and Number of NF Stages = 4)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P0ϕ [MHz]			
	100			
	RIICnMR1.CKS[2:0]	RIICnBRH.BRH	RIICnBRL.BRL	Actual transfer rate [kbps]
12	111b	25 (H’F9)	28 (H’FC)	12
50	101b	23 (H’F7)	28 (H’FC)	49.6
100	100b	24 (H’F8)	27 (H’FB)	99.2
400	010b	25 (H’F9)	26 (H’FA)	396.8
1000	001b	14 (H’EE)	24 (H’F8)	1000

26.3.14 RIICnDRT — I^{2}C Bus Transmit Data Register

Access Size:	RIICnDRT is a 32-bit readable/writable register. RIICnDRTL and RIICnDRTH are 16-bit readable/writable registers. RIICnDRTLL, RIICnDRTLH, RIICnDRTHL, and RIICnDRTHH are 8-bit readable/writable registers.
Address(es):	RIICnDRT: <RIICn_base> + H’003C RIICnDRTL: <RIICn_base> + H’003C, RIICnDRTH: <RIICn_base> + H’003E RIICnDRTLL: <RIICn_base> + H’003C, RIICnDRTLH: <RIICn_base> + H’003D, RIICnDRTHL: <RIICn_base> + H’003E, RIICnDRTHH: <RIICn_base> + H’003F
Initial Value:	H’0000 00FF This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When RIICnDRT detects a space in the I^{2}C bus shift register (RIICnDRS), it transfers the transmit data that has been written to RIICnDRT to RIICnDRS and starts transmitting data in transmit mode.

The double-buffer structure of RIICnDRT and RIICnDRS allows continuous transmit operation if the next transmit data has been written to RIICnDRT while the RIICnDRS data is being transmitted.

RIICnDRT can always be read and written. Write transmit data to RIICnDRT once when a transmit data empty interrupt (INTRIICTI) request is generated. When writing to bits 8 to 15, be sure to write 0 to these bits.

26.3.15 RIICnDRR — I^{2}C Bus Receive Data Register

Access Size: RIICnDRR is a 32-bit readable/writable register.
RIICnDRRL and RIICnDRRH are 16-bit readable/writable registers.
RIICnDRRLL, RIICnDRRLH, RIICnDRRHL, and RIICnDRRHH are 8-bit readable/writable registers.

Address(es): RIICnDRR: <RIICn_base> + H’0040
RIICnDRRL: <RIICn_base> + H’0040, RIICnDRRH: <RIICn_base> + H’0042
RIICnDRRLL: <RIICn_base> + H’0040, RIICnDRRLH: <RIICn_base> + H’0041,
RIICnDRRHL: <RIICn_base> + H’0042, RIICnDRRHH: <RIICn_base> + H’0043

Initial Value: H’0000 0000 This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

When 1 byte of data has been received, the received data is transferred from the I^{2}C bus shift register (RIICnDRS) to RIICnDRR to enable the next data to be received.

The double-buffer structure of RIICnDRS and RIICnDRR allows continuous receive operation if the received data has been read from RIICnDRR while RIICnDRS is receiving data.

RIICnDRR cannot be written. Read data from RIICnDRR once when a receive data full interrupt (INTRIICRI) request is generated.

If DRR receives the next receive data before the current data is read from RIICnDRR (while the RIICnSR2.RDRF flag is 1), the RIIC automatically holds the SCL clock low one cycle before the RDRF flag is set to 1 next.

26.3.16 RIICnDRS — I^{2}C Bus Shift Register

Access Size: This register is not accessible.
Address(es): —
Initial Value: H'0000 00FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRS[7:0]							
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

RIICnDRS is an 8-bit shift register to transmit and receive data.

During transmission, transmit data is transferred from RIICnDRT to RIICnDRS and is sent from the SDA pin. During reception, data is transferred from RIICnDRS to RIICnDRR after 1 byte of data has been received.

RIICnDRS cannot be accessed directly.

26.4 Interrupt Sources

The RIIC issues eight types of interrupt request: transmit end, receive data full, transmit data empty, stop condition detection, start condition detection, NACK reception, arbitration-lost, and timeout.

Table 26.11 lists details of the several interrupt requests. The receive data full and transmit data empty sources are both capable of launching data transfer by the DMAC.

Table 26.11 Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	DMAC Launching	Priority*1	Interrupt Condition
INTRIICRI	Receive-data-full	RDRF	Possible	High	RDRF = 1 • RIE = 1
INTRIICTI	Transmit-data-empty	TDRE	Possible	↑	TDRE = 1 • TIE = 1
INTRIICTEI	Transmission complete	TEND	Not possible		TEND = 1 • TEIE = 1
INTRIICNAKI	Reception of a NACK	NACKF	Not possible		NACKF = 1 • NAKIE = 1
INTRIICSPI	Detection of a stop condition	STOP	Not possible		STOP = 1 • SPIE = 1
INTRIICSTI	Detection of a start condition	START	Not possible		START = 1 • STIE = 1
INTRIICALI	Arbitration lost	AL	Not possible		AL = 1 • ALIE = 1
INTRIICTMOI	Timeout	TMOF	Not possible	Low	TMOF = 1 • TMOIE = 1

Note 1. When the interrupt priority register (ICDIPRn) setting is the same

Clear or mask the each flag during interrupt handling.

CAUTIONS

1. There is a latency (delay) between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. Thus, when an interrupt flag has been cleared or masked, read the relevant flag again to check whether clearing or masking has been completed, and then return from interrupt processing. Returning from interrupt processing without checking that writing to the module has been completed creates a possibility of repeated processing of the same interrupt.
2. Since INTRIICRI and INTRIICTI are edge-detected interrupts, they do not require clearing.
3. When using the INTRIICTEI interrupt, clear the RIICnSR2.TEND flag in the INTRIICTEI interrupt processing.
4. When using the INTRIICSPI interrupt, clear the RIICnSR2.STOP flag in the INTRIICSPI interrupt processing.
5. When using the INTRIICSTI interrupt, clear the RIICnSR2.START flag in the INTRIICSTI interrupt processing.
6. When using the INTRIICNAKI interrupt, clear the RIICnSR2.NACKF flag in the INTRIICNAKI interrupt processing.
7. When using the INTRIICALI interrupt, clear the RIICnSR2.AL flag in the INTRIICALI interrupt processing.
8. When using the INTRIICTMOI interrupt, clear the RIICnSR2.TMOF flag in the INTRIICTMOI interrupt processing.

26.5 Operation

26.5.1 Communication Data Format

The I²C bus format consists of 8-bit data and 1-bit acknowledge (one frame). After a start condition or restart condition is issued, the master device sends the slave address and data direction in the first frame. The specified slave is valid until a stop condition is issued or a new slave is specified by a restart condition.

Figure 26.3 shows the I²C bus format, and **Figure 26.4** shows the I²C bus timing.

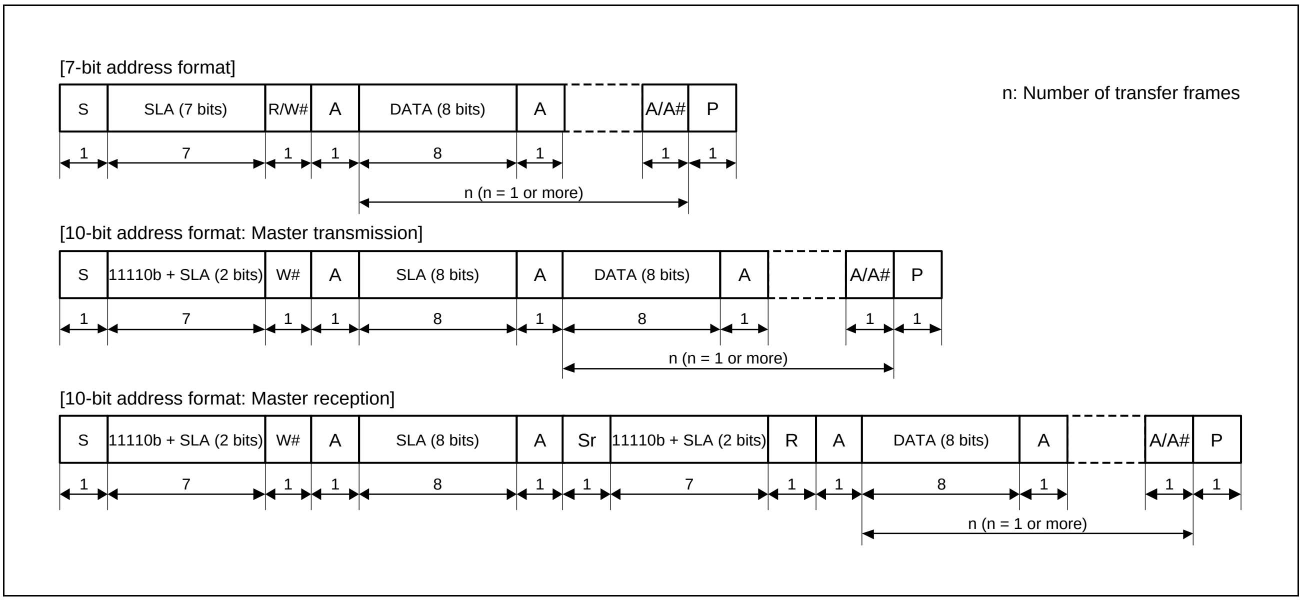
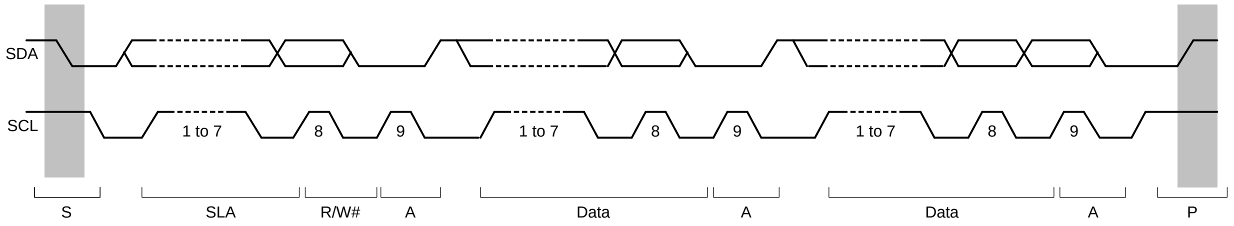


Figure 26.3 I²C Bus Format



S: Start condition. The master device drives the SDA line low from high level while the SCL line is at a high level.

SLA: Slave address, by which the master device selects a slave device.

R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.

A: Acknowledge. The receive device drives the SDA line low. (In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.)

A#: Not-acknowledge. The receiving device has not returned a response or is not present so the SDA line has remained at the high level.

Sr: Restart condition. The master device drives the SDA line low from the high level after the setup time has elapsed with the SCL line at the high level.

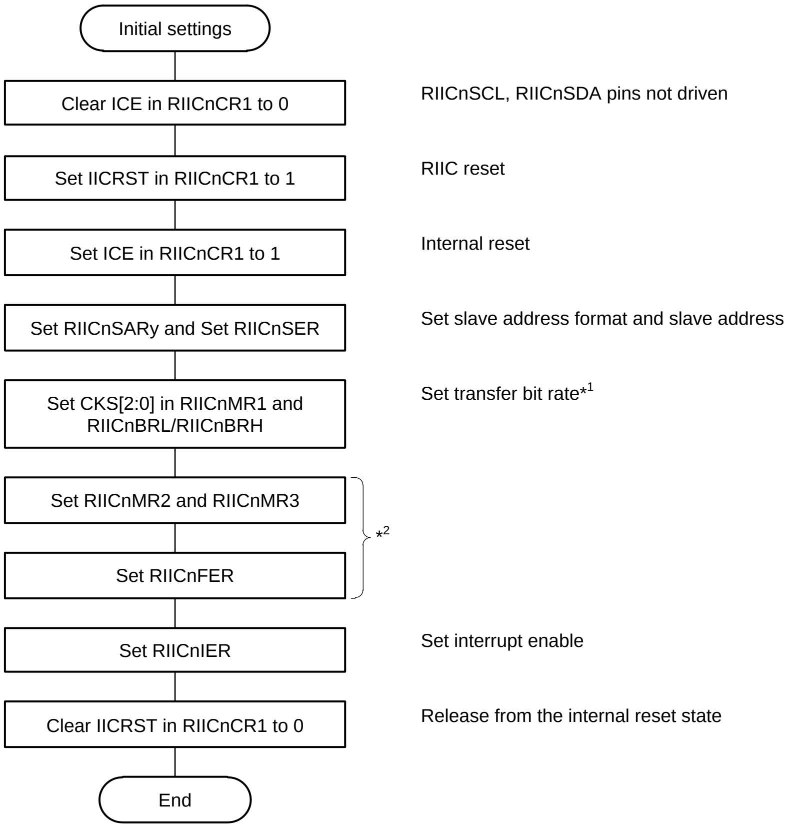
DATA: Transmitted or received data

P: Stop condition. The master device drives the SDA line high from low level while the SCL line is at a high level.

Figure 26.4 I^{2}C Bus Timing (SLA = 7 Bits)

26.5.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC according to the procedure in **Figure 26.5**.



Note: y = 0 to 2

Note 1. When the RIIC is used only in slave mode, set the RIICnBRL register to a value at least the same as the data setup time.

Note 2. Set these registers as necessary.

Figure 26.5 Example of RIIC Initialization Flowchart

26.5.3 Master Transmit Operation

In master transmit operation, the RIIC outputs the SCL (clock) and transmitted data signals as the master device, and the slave device returns acknowledgements. **Figure 26.6** shows an example of usage of master transmission and **Figure 26.7** to **Figure 26.9** show the timing of operations in master transmission.

The following describes the procedure and operations for master transmission.

(1) Set the RIICnCR1.IICRST bit 1 to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSARy, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see **Figure 26.5**). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.

(2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. At the same time, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the internal levels for the SDA output state and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.

(3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the slave address and the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, the RIIC continues in master transmit mode.

Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnCR2.SP bit to issue a stop condition.

For data transmission with an address in the 10-bit format, start by writing 1111 0b, the two higher-order bits of the slave address, and W# to RIICnDRT as the first address transmission. Then, as the second address transmission, write the eight lower-order bits of the slave address to RIICnDRT.

(4) After confirming that the RIICnSR2.TDRE flag is 1, write the data for transmission to the RIICnDRT register. The RIIC automatically holds the SCL line low until the data for transmission are ready or a stop condition is issued.

(5) After the last byte of the data to be transmitted is written to the RIICnDRT register, wait until the value of the RIICnSR2.TEND flag returns to 1, and then set the RIICnCR2.SP bit to 1 (stop condition issuance request). Upon receiving a stop condition issuance request, the RIIC issues the stop condition.

(6) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00b and enters slave receive mode. Furthermore, it automatically clears the RIICnSR2.TDRE and TEND flags to 0, and sets the RIICnSR2.STOP flag in to 1.

(7) Clear the RIICnSR2.NACKF and STOP flags to 0.

CAUTION

Operations for transfer start if the RIICnSR2.NACKF flag is cleared to 0 before RIICnSR2.STOP is set to 1. Be sure to confirm that RIICnSR2.STOP is set to 1 before clearing RIICnSR2.NACKF to 0. In particular, when the NACK receive interrupt (INTRIICNAKI) is in use, take care not to clear the NACKF flag to 0 before the STOP flag is set to 1 during interrupt processing.

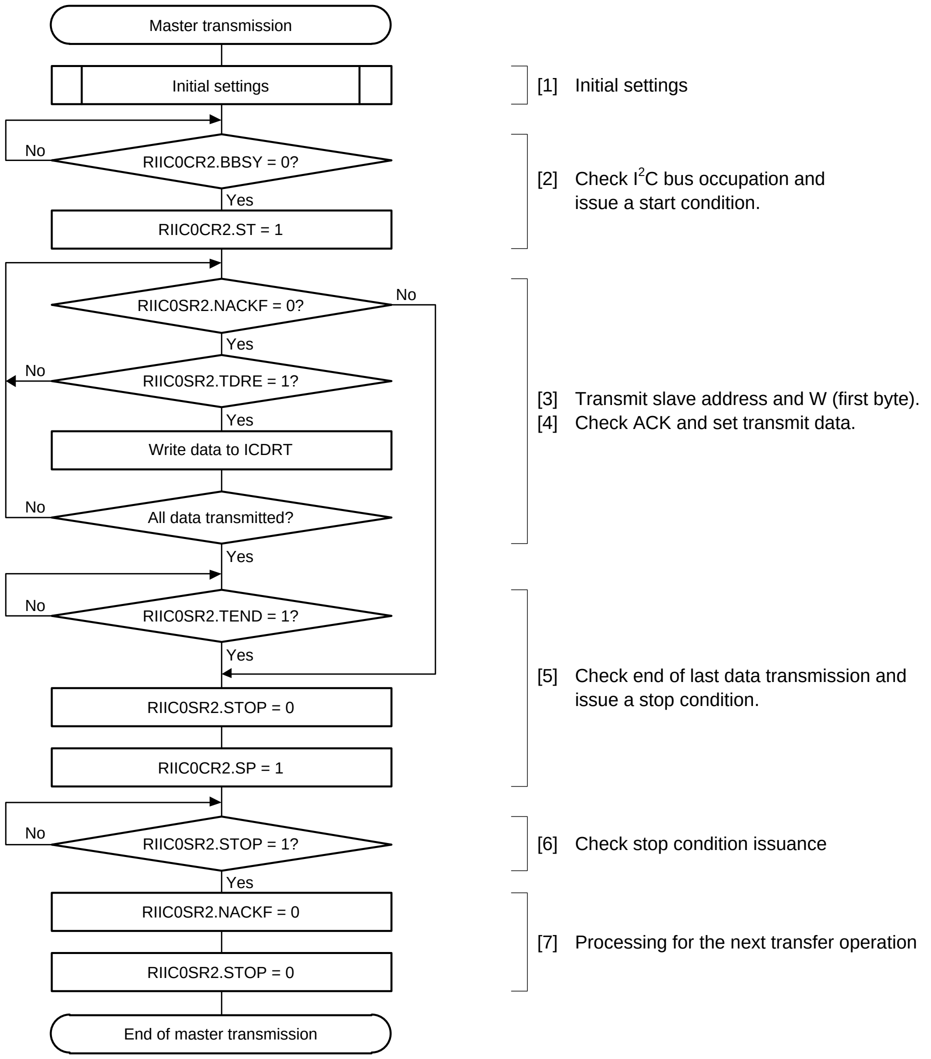


Figure 26.6 Example of Master Transmission Flowchart

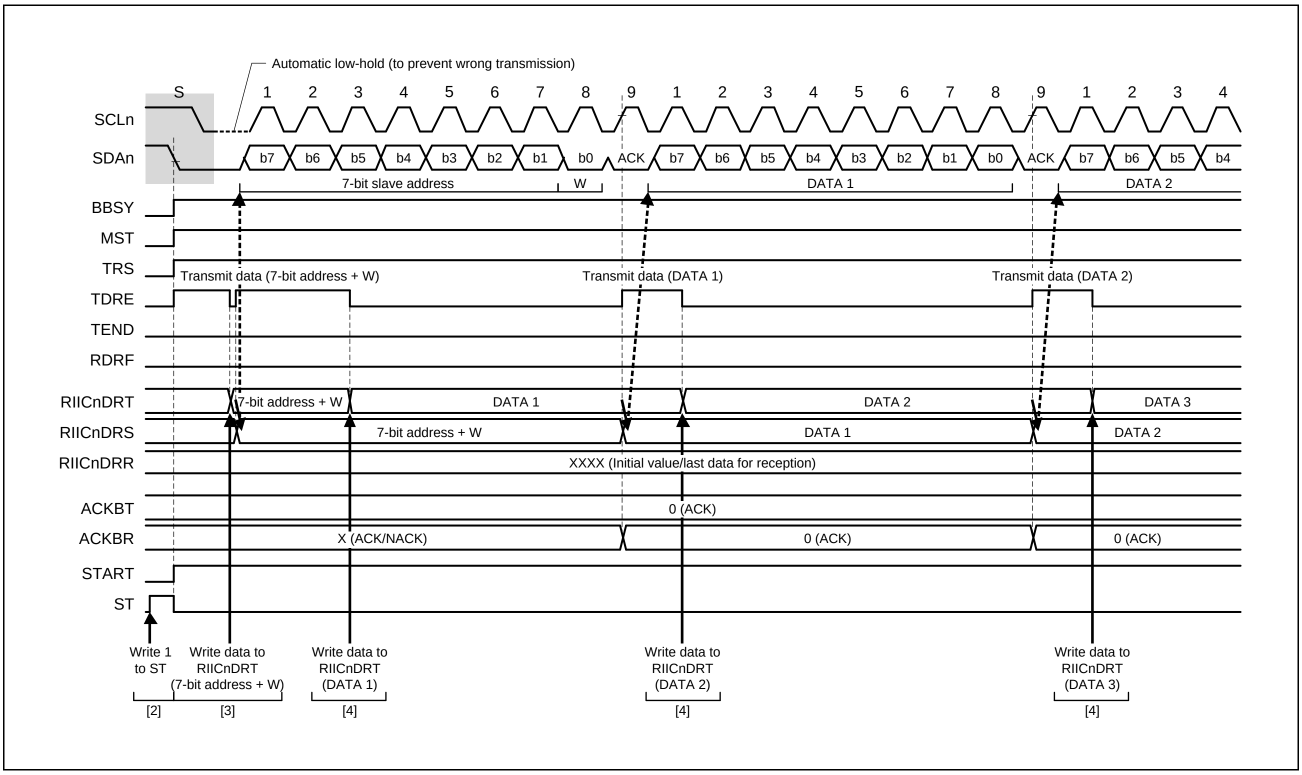


Figure 26.7 Master Transmit Operation Timing (1) (7-Bit Address Format)

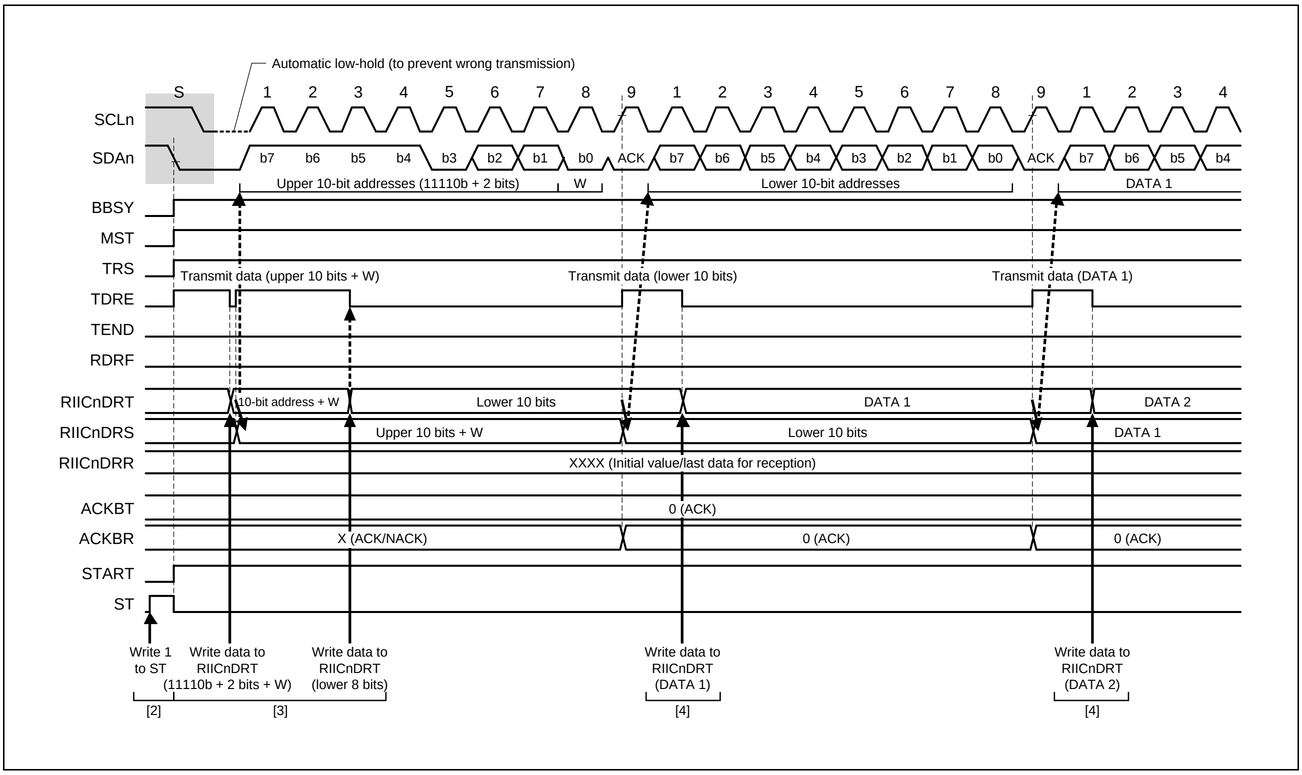


Figure 26.8 Master Transmit Operation Timing (2) (10-Bit Address Format)

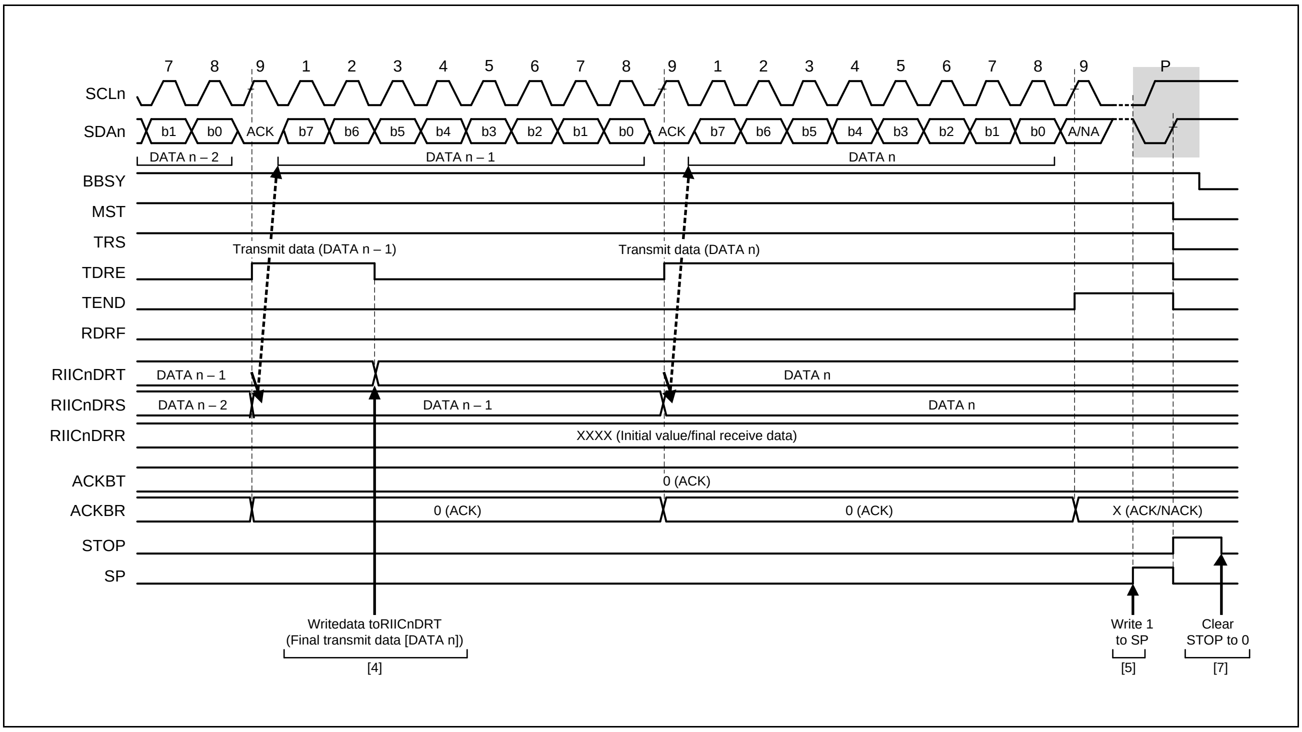


Figure 26.9 Master Transmit Operation Timing (3)

26.5.4 Master Receive Operation

In master receive operation, the RIIC as a master device outputs the SCL (clock) signal, receives data from the slave device, and returns acknowledgements. Since the RIIC must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

Figure 26.10 shows an example of usage for the master reception of 3 or more bytes (7-bit address format), **Figure 26.14** shows an example of usage for the master reception of 1 or 2 bytes (7-bit address format), and **Figure 26.11** to **Figure 26.13** show the timing of operations in master reception.

The following describes the procedure and operations for master reception.

(1) Set the RIICnCR1.IICRST bit to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSARy, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see **Figure 26.5**). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.

(2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. When the RIIC detects the start condition, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the levels for the SDA output and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.

(3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the RIICnCR2.TRS bit is automatically updated to select transmit or receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the RIICnCR2.TRS bit is cleared to 0 on the rising edge of the ninth cycle of SCL (the clock signal), placing the RIIC in master receive mode. At this time, the TDRE flag is automatically cleared to 0 and the RIICnSR2.RDRF flag is automatically set to 1.

Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnCR2.SP bit to issue a stop condition.

For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a restart condition. After that, transmitting 1111 0b, the two higher-order bits of the slave address, and the R bit places the RIIC in master receive mode.

(4) Dummy read RIICnDRR after confirming that the RIICnSR2.RDRF flag is 1; this makes the RIIC start output of the SCL (clock) signal and start data reception.

(5) After 1 byte of data has been received, the RIICnSR2.RDRF flag is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the RIICnMR3.RDRFS bit. Reading out RIICnDRR at this time will produce the received data, and the RDRF flag is automatically cleared to 0 at the same time. Furthermore, the value of the acknowledgement field received during the ninth cycle of SCL clock is returned as the value set in the RIICnMR3.ACKBT bit. Furthermore, if the next byte to be received is the next to last byte, set the RIICnMR3.WAIT bit to 1 (for wait insertion) before reading the RIICnDRR (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the RIICnMR3.ACKBT bit

to 1 (NACK) in step (6), due to other interrupts, etc., this fixes the SCL line to the low level on the rising edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a stop condition is possible.

(6) When the RIICnMR3.RDRFS bit is 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the RIICnMR3.ACKBT bit to 1 (NACK).

(7) After reading out the byte before last from the RIICnDRR register, if the value of the RIICnSR2.RDRF flag is confirmed to be 1, write 1 to the RIICnCR2.SP bit (stop condition issuance request) and then read the last byte from RIICnDRR. When RIICnDRR is read, the RIIC is released from the wait state and issues the stop condition after low-level output in the ninth clock cycle is completed or the SCL line is released from the low-hold state.

(8) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00b and enters slave receive mode. Furthermore, detection of the stop condition leads to setting of the RIICnSR2.STOP flag to 1.

(9) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

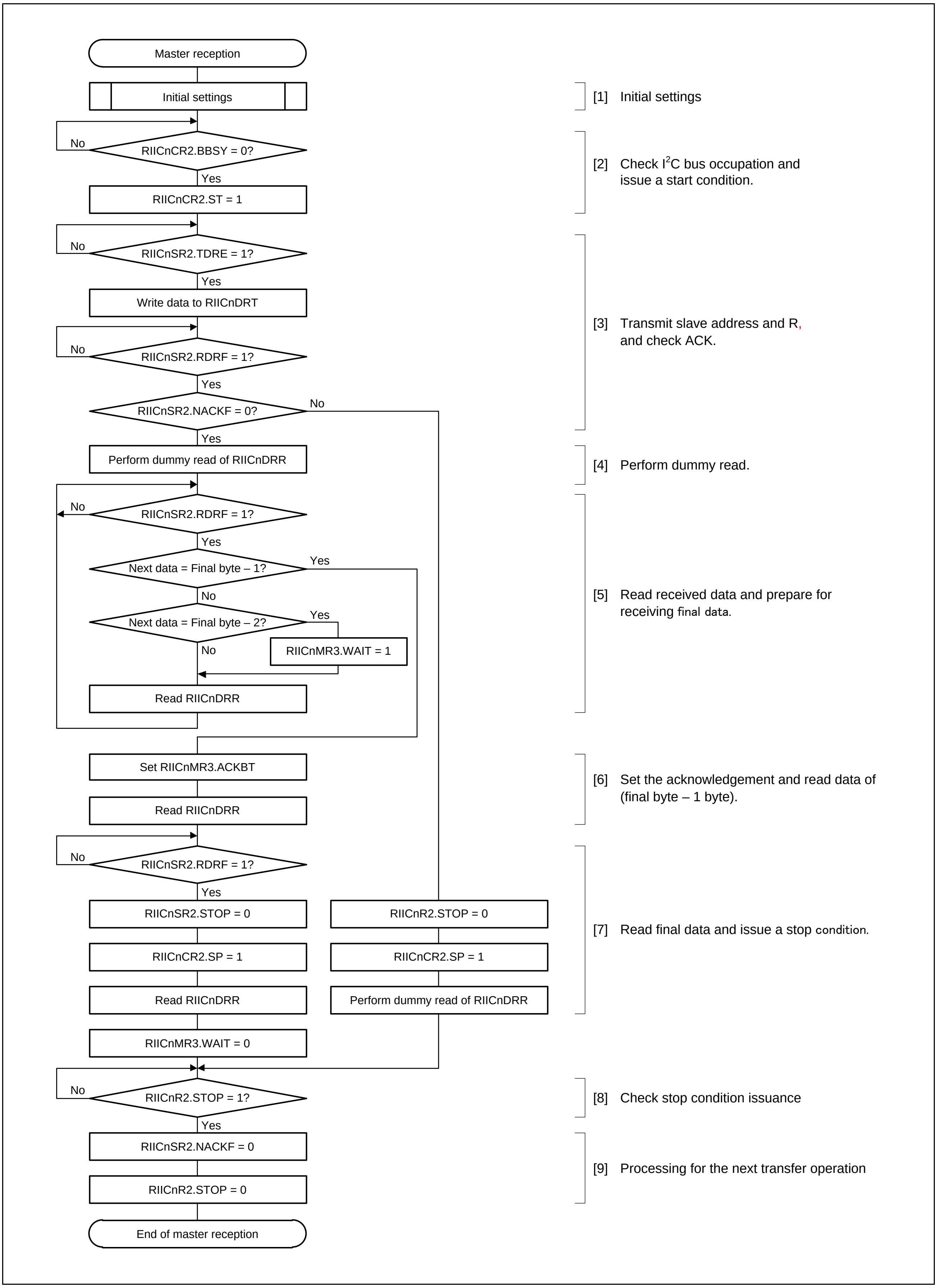


Figure 26.10 Example Flowchart for the Master Reception of 3 or More Bytes (7-Bit Address Format)

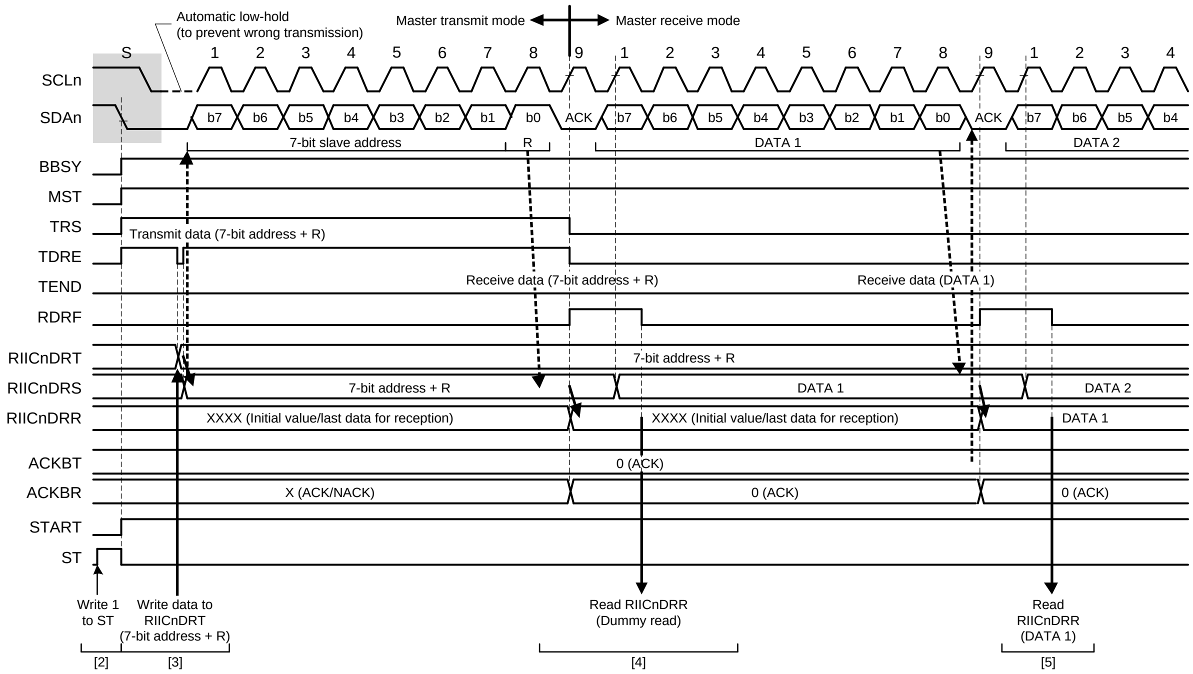


Figure 26.11 Master Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

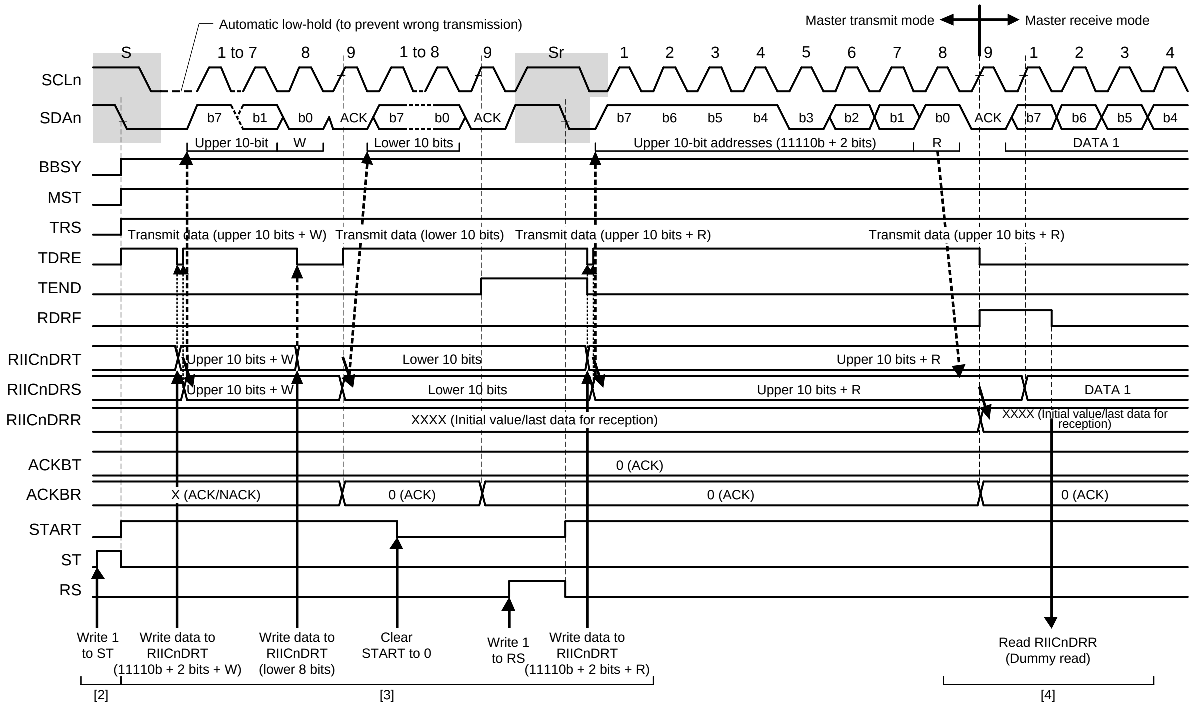


Figure 26.12 Master Receive Operation Timing (2) (10-Bit Address Format, when RDRFS = 0)

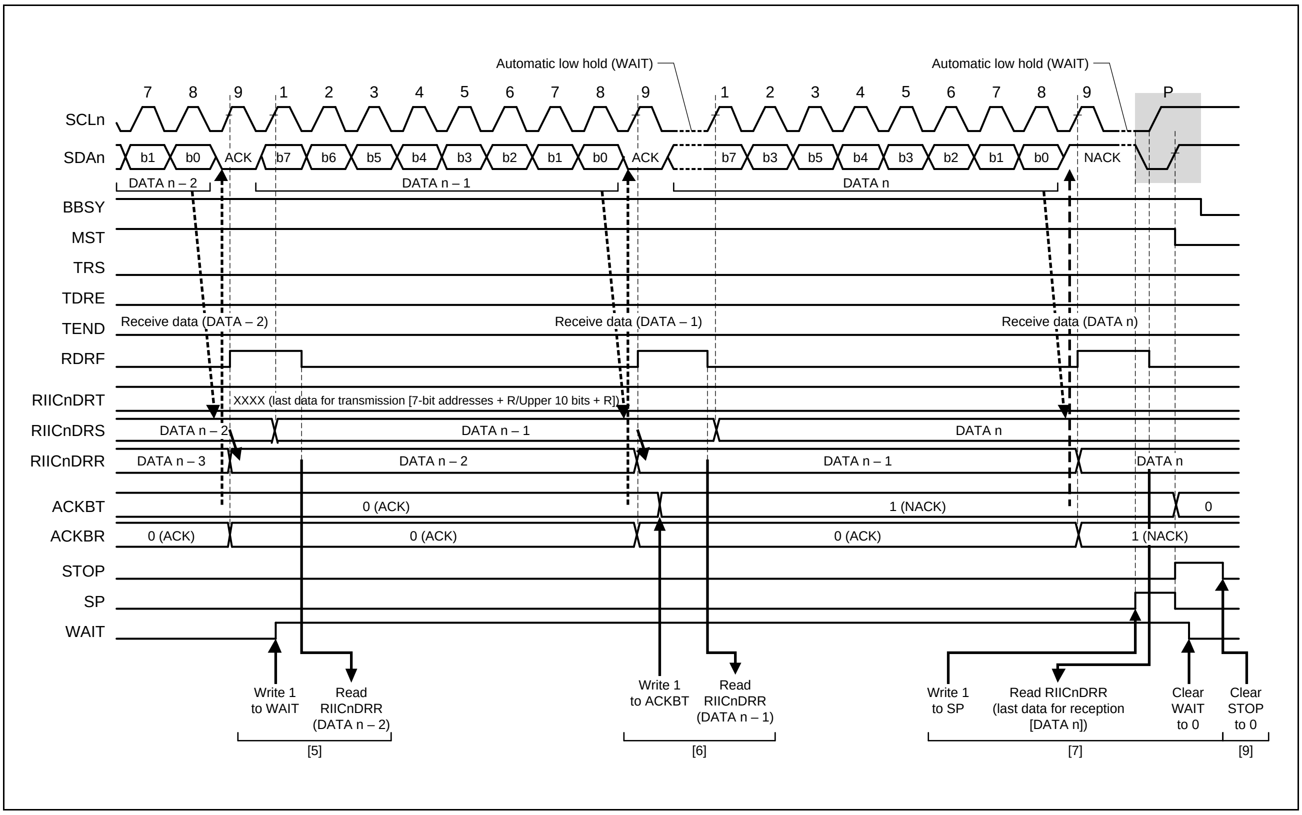


Figure 26.13 Master Receive Operation Timing (3) (when RDRFS = 0)

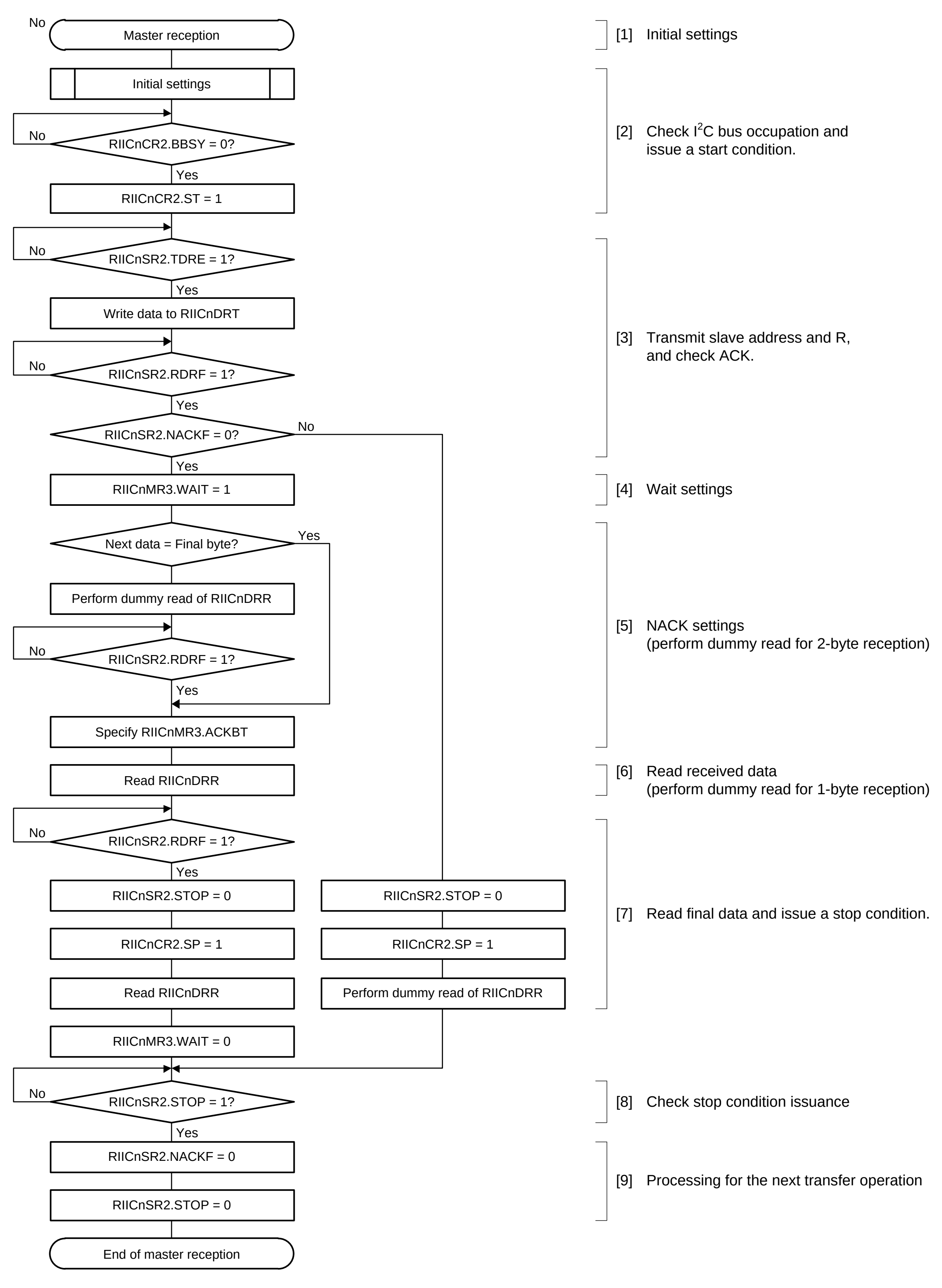


Figure 26.14 Example Flowchart for the Master Reception of 1 or 2 Bytes (7-Bit Address Format)

26.5.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL (clock) signal, the RIIC transmits data as a slave device, and the master device returns acknowledgements.

Figure 26.15 shows an example of usage of slave transmission and **Figure 26.16** and **Figure 26.17** show the timing of operations in slave transmission.

The following describes the procedure and operations for slave transmission.

(1) Follow the procedure in **Figure 26.5** to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmit mode by setting both the RIICnCR2.TRS bit and the RIICnSR2.TDRE flag to 1.

(3) After the RIICnSR2.TEND flag is confirmed to be 1, write the data for transmission to the RIICnDRT register. At this time, if the RIIC receives no acknowledge from the master device (receives an NACK signal) while the RIICnFER.NACKE bit is 1, the RIIC suspends transfer of the next data.

(4) Wait unit the RIICnSR2.TEND flag is set to 1 while the RIICnSR2.TDRE flag is 1, after the RIICnSR2.NACKF flag is set to 1 or the last byte for transmission is written to the RIICnDRT register. When the RIICnSR2.NACKF flag or the TEND flag is 1, the RIIC drives the SCL line low on the ninth falling edge of SCL clock.

(5) When the RIICnSR2.NACKF flag or the RIICnSR2.TEND flag is 1, dummy read RIICnDRR to complete the processing. This releases the SCL line.

(6) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2), flags RIICnSR2.TDRE and TEND, and the RIICnCR2.TRS bit to 0, and enters slave receive mode.

(7) Clear the RIICnSR2.NACKF and STOP flags to 0.

CAUTION

Operations for transfer start if the RIICnSR2.NACKF flag is cleared to 0 before RIICnSR2.STOP is set to 1. Be sure to confirm that RIICnSR2.STOP is set to 1 before clearing RIICnSR2.NACKF to 0. In particular, when the NACK receive interrupt (INTRIICNAKI) is in use, take care not to clear the NACKF flag to 0 before the STOP flag is set to 1 during interrupt processing.

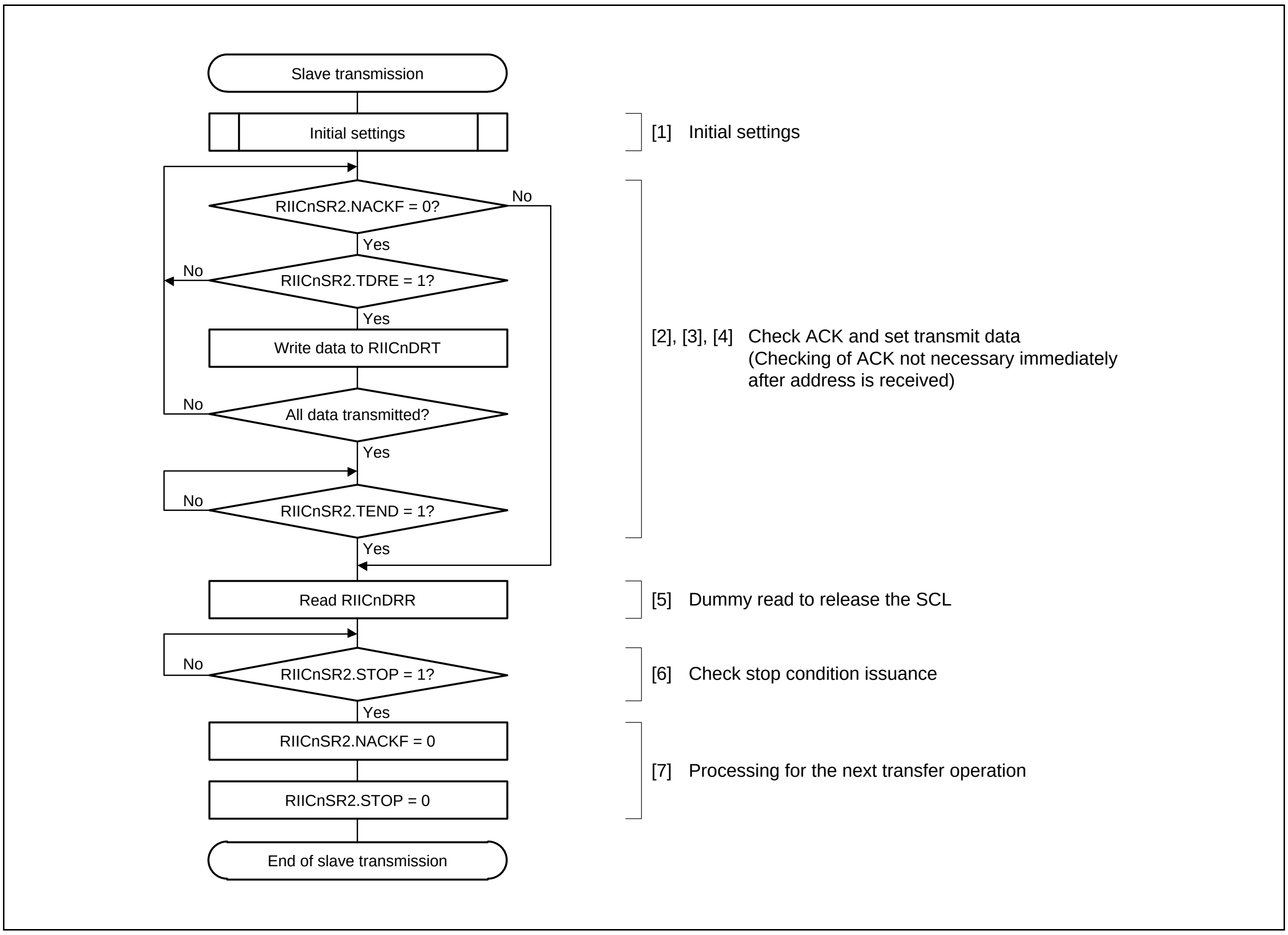


Figure 26.15 Example of Slave Transmission Flowchart

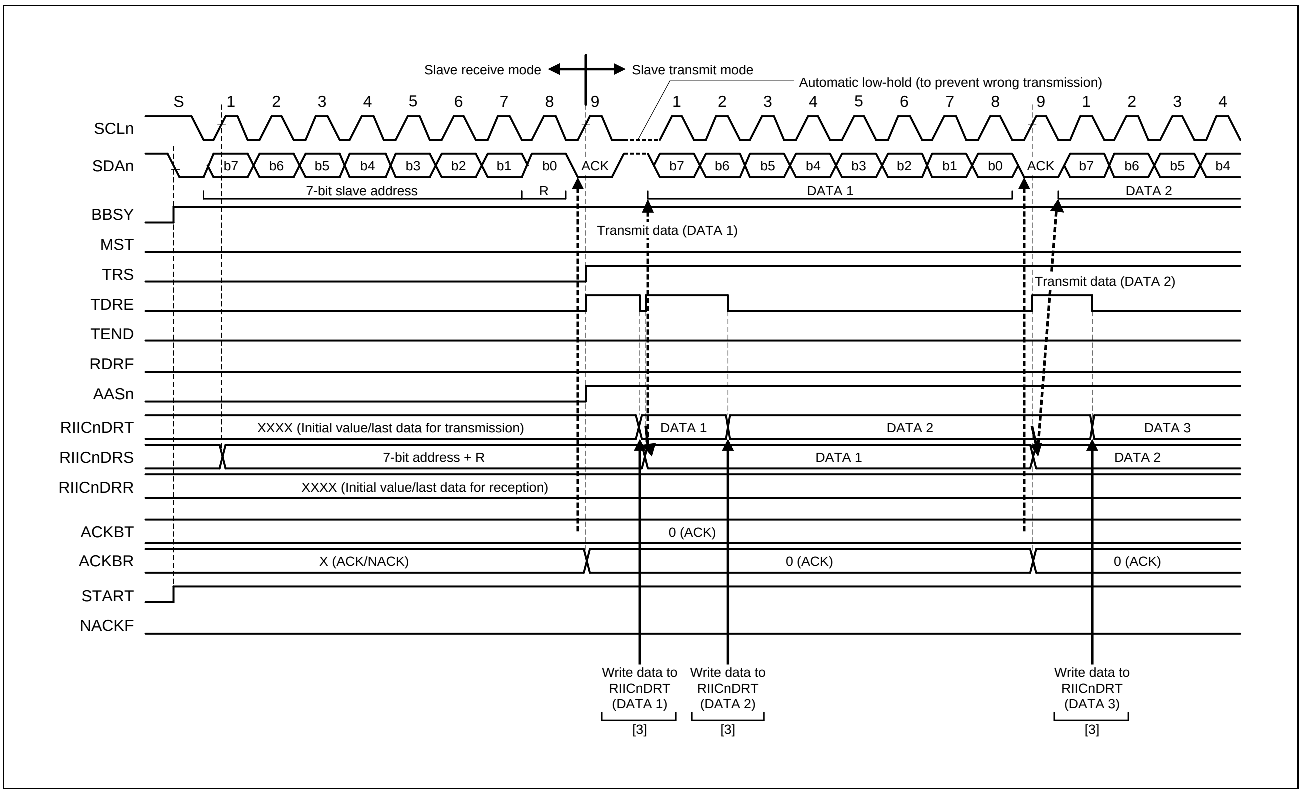


Figure 26.16 Slave Transmit Operation Timing (1) (7-Bit Address Format)

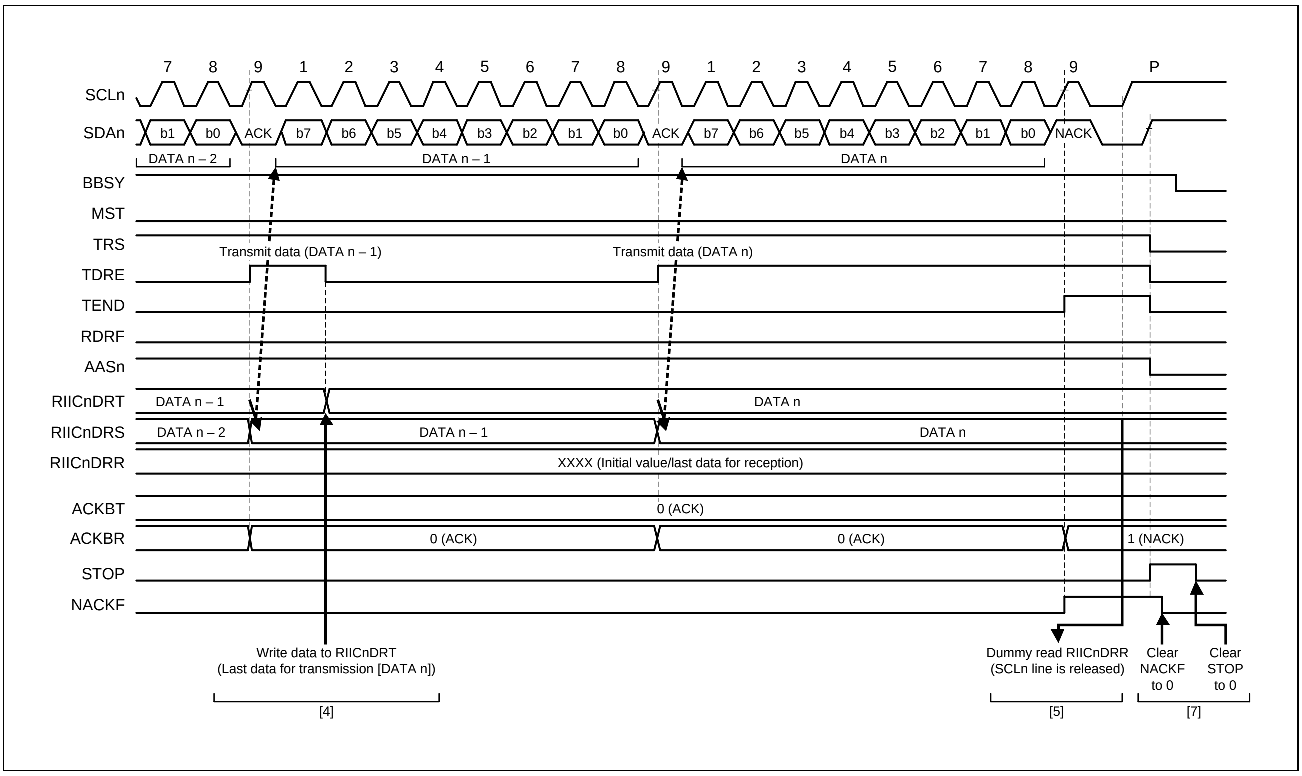


Figure 26.17 Slave Transmit Operation Timing (2)

26.5.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL clock and transmit data, and the RIIC returns acknowledgements as a slave device.

Figure 26.18 shows an example of usage of slave reception and **Figure 26.19** and **Figure 26.20** show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

(1) Follow the procedure in **Figure 26.5** to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0, the RIIC continues to place itself in slave receive mode and sets the RIICnSR2.RDRF flag to 1.

(3) After the RIICnSR2.STOP flag is confirmed to be 0 and the RIICnSR2.RDRF flag to be 1, dummy read RIICnDRR (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower eight bits when the 10-bit address format is selected).

(4) When RIICnDRR is read, the RIIC automatically clears the RIICnSR2.RDRF flag to 0. If reading of RIICnDRR is delayed and a next byte is received while the RDRF flag is still set to 1, the RIIC holds the SCL line low from one SCL cycle before the timing with which RDRF should be set. In this case, reading RIICnDRR releases the SCL line from being held at the low level.

When the RIICnSR2.STOP flag is 1 and the RIICnSR2.RDRF flag is also 1, read RIICnDRR until all the data is completely received.

(5) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 0.

(6) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.STOP flag to 0 for the next transfer operation.

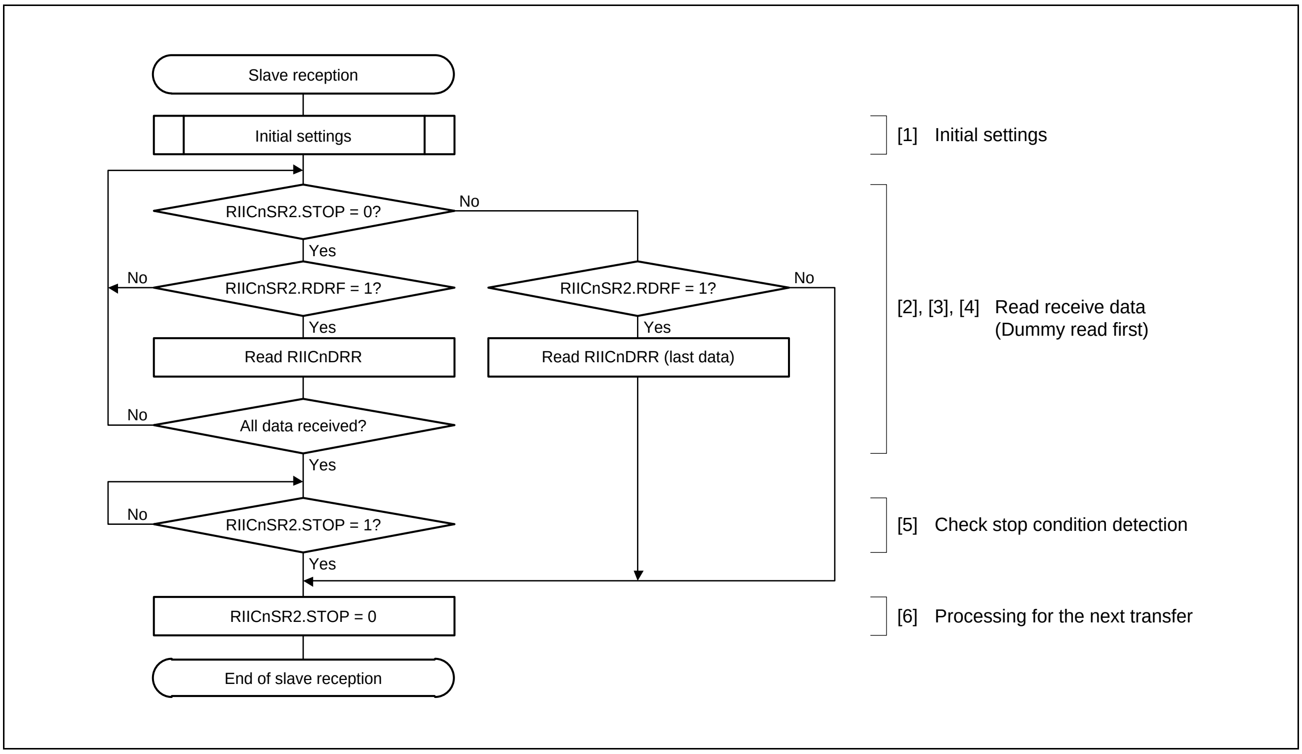


Figure 26.18 Example of Slave Reception Flowchart

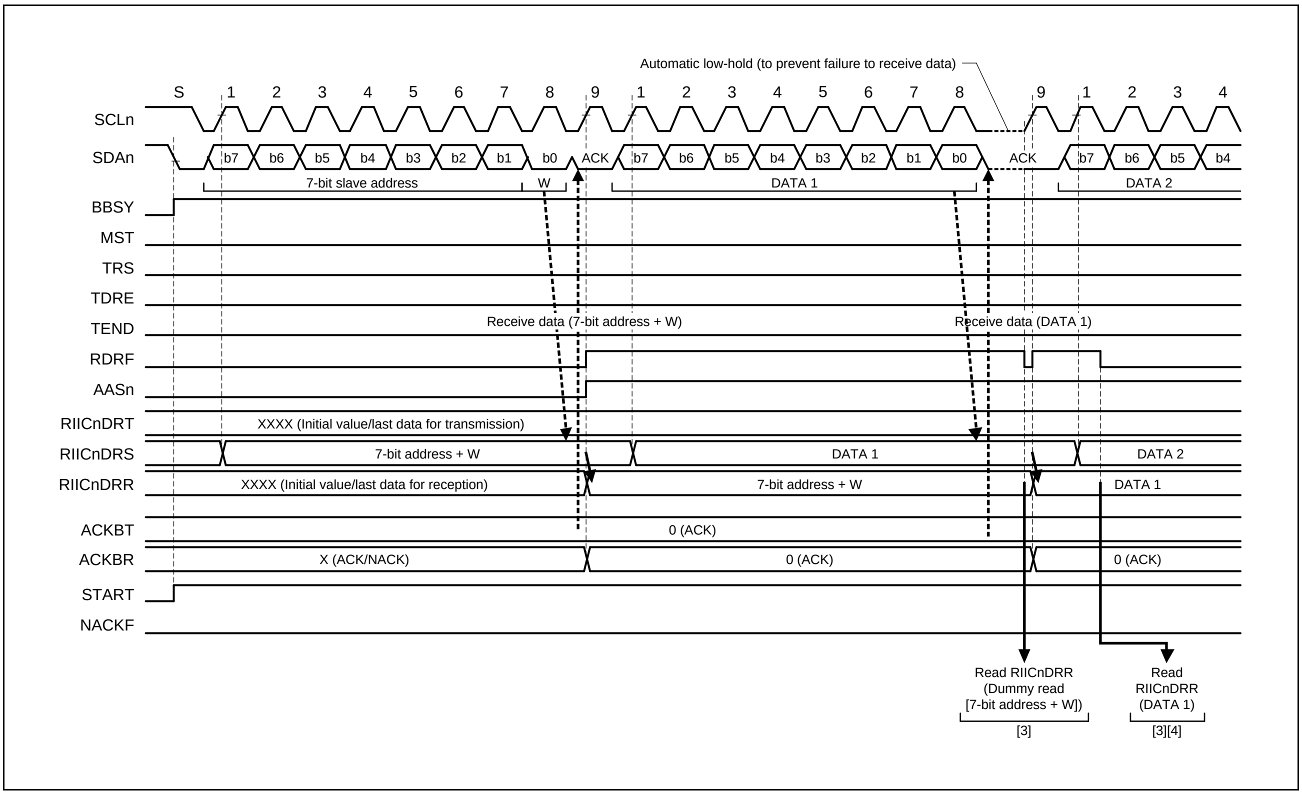


Figure 26.19 Slave Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

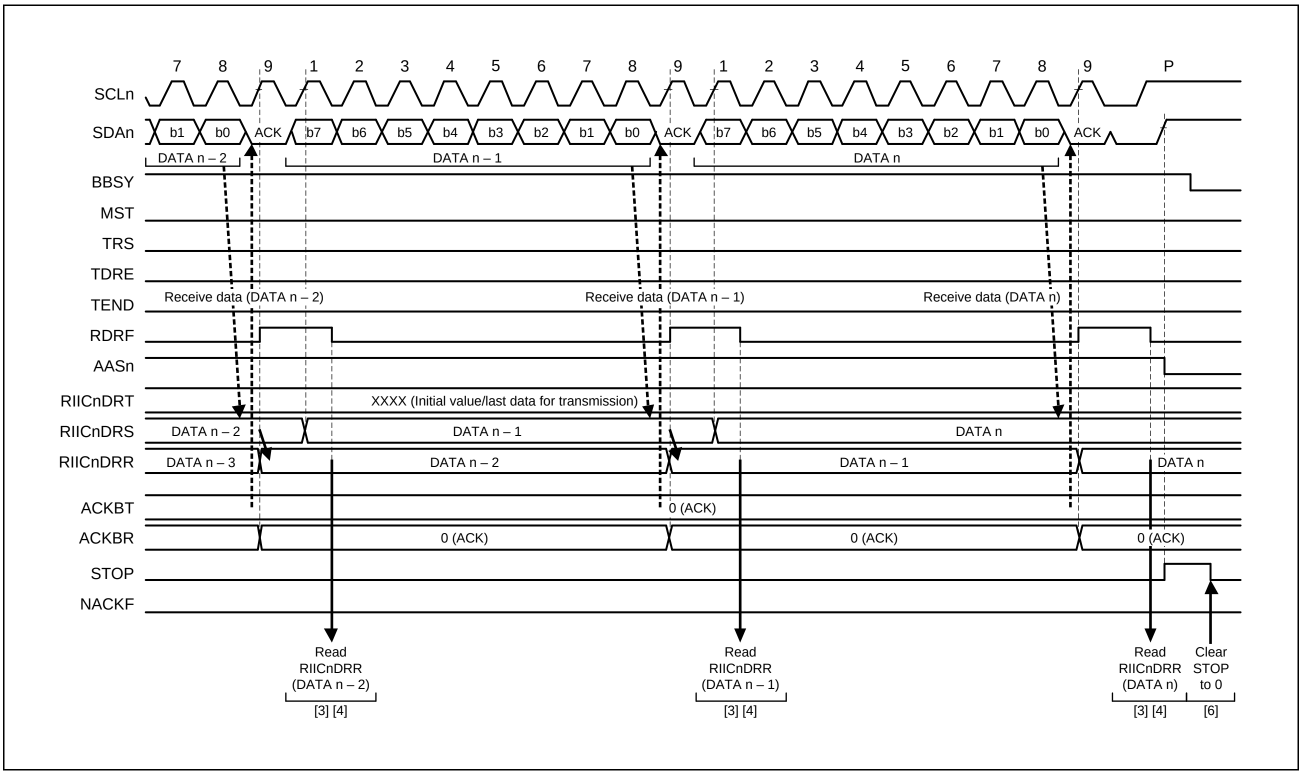


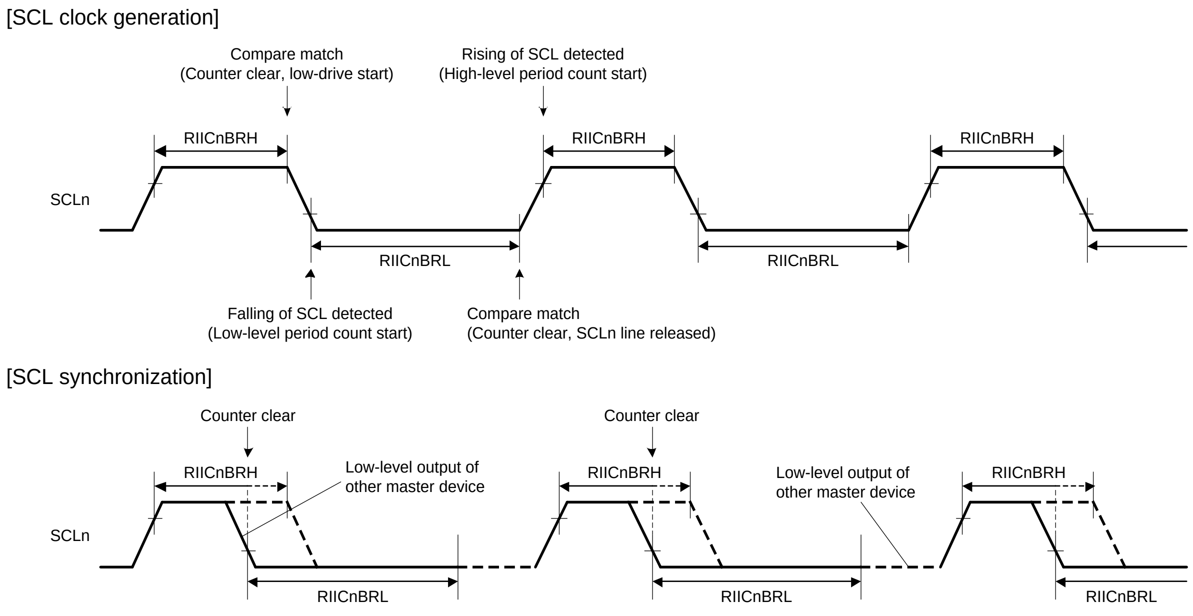
Figure 26.20 Slave Receive Operation Timing (2) (when RDRFS = 0)

26.6 SCL Synchronization Circuit

In generation of the SCL (clock) signal, the RIIC starts counting out the value for width at high level specified in RIICnBRH when it detects a rising edge on the SCL line and drives the SCL line low once counting of the width at high level is complete. When the RIIC detects the falling edge of the SCL line, it starts counting out the width at low level period specified in RIICnBRL, and then stops driving the SCL line (releases the line) once counting of the width at low level is complete. The SCL (clock) signal is thus generated.

If multiple master devices are connected to the I²C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, the RIIC is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCL line during communication.

When the RIIC has detected a rising edge on the SCL line and thus started counting out the width at high level specified in RIICnBRH, and the level on the SCL line falls because an SCL signal is being generated by another master device, the RIIC stops counting when it detects the falling edge, drives the level on the SCL line low, and starts counting out the width at low level specified in RIICnBRL. When the RIIC finishes counting out the width at low level, it stops driving the SCL line to the low level (i.e. releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in the RIIC, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCL line has been released. When the RIIC finishes outputting the low-level period of the SCL clock, the SCL line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the RIICnFER.SCLE bit is set to 1.



Note: RIICnBRH: I²C bus bit rate high-level register (SCL clock high-level period counter)
RIICnBRL: I²C bus bit rate low-level register (SCL clock low-level period counter)

Figure 26.21 Generation and Synchronization of the SCL Signal from the RIIC

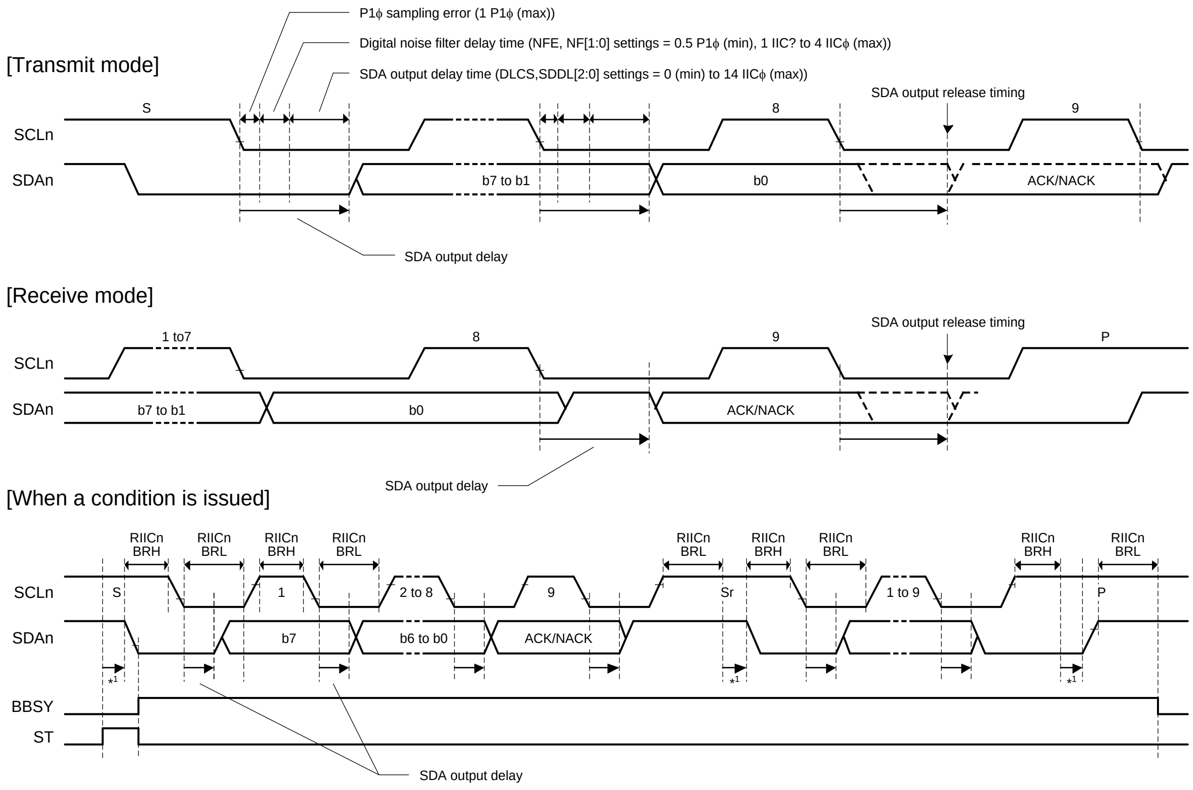
26.7 Facility for Delaying SDA Output

The RIIC module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the start, restart, and stop conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL (clock) signal is at the low level. Doing this leads to usage with the aim of preventing erroneous operation of communications devices, with the aim of satisfying the 300-ns (min.) data-hold time requirement of the SMBus specification.

The output delay facility is enabled by setting the RIICnMR2.SDDL[2:0] bits to any value other than 000b, and disabled by setting the same bits to 000b.

While the SDA output delay facility is enabled (i.e. while the SDDL[2:0] bits in IMCR2 are set to any value other than 000b), the RIICnMR2.DLCS bit selects the clock source for counting by the SDA output delay counter as the internal base clock (IICϕ) for the RIIC module or as a clock signal derived by dividing the frequency of the internal base clock by two (IICϕ/2). The counter counts the number of cycles set in the SDDL[2:0] bits in IMCR2. After counting of the set number of cycles of delay is completed, the RIIC module places the required output (start, restart, or stop condition, data, or an ACK or NACK signal) on the SDA line.



Note 1. The output is delayed by the number of cycles set by the SDDL [2:0] bits when a start (S), restart (Sr), or stop (P) condition is issued.

Figure 26.22 SDA Output Delay Facility

26.8 Digital Noise-Filter Circuits

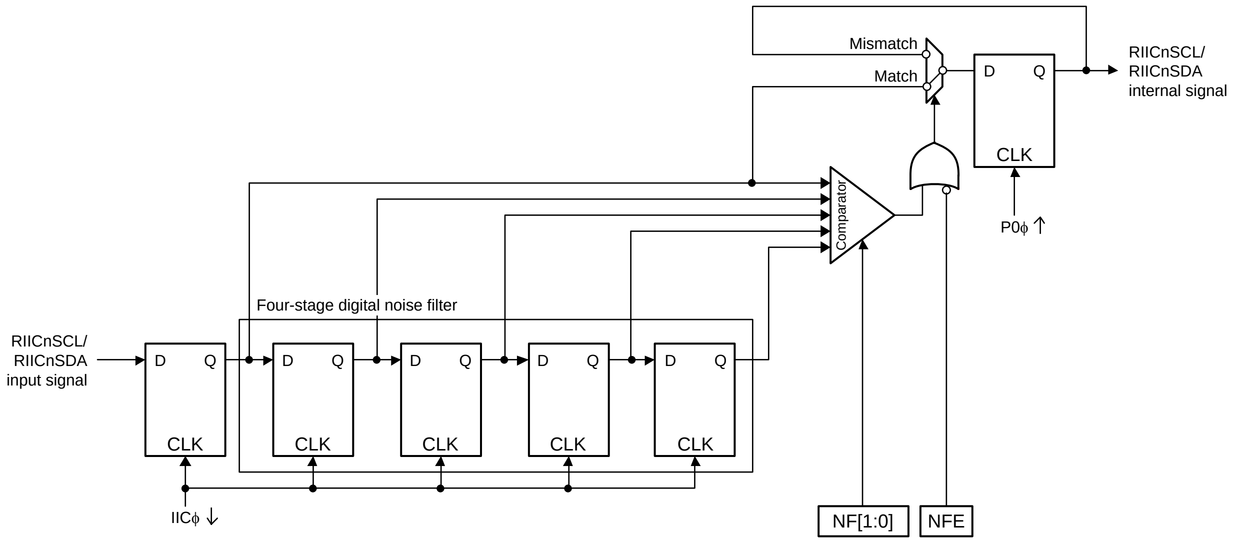
Figure 26.23 is a block diagram of the digital noise-filter circuit. When the NFE bit in the RIICnFER register is set to 1, input to the RIICnSCL and RIICnSDA pins are conveyed to the internal circuitry through digital noise-filter circuits.

The on-chip digital noise-filter circuit of the RIIC consists of four flip-flop circuit stages connected in series and a match- detection circuit.

The number of effective stages in the digital noise filter is selected by the RIICnMR3.NF[1:0] bits. The selected number of effective stages determines the noise-filtering capability as a period from one to four IICϕ cycles.

The input signal to the RIICnSCL pin (or RIICnSDA pin) is sampled on falling edges of the IICϕ signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the RIICnMR3.NF[1:0] bits, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between P0ϕ and IICϕ is small when the RIICnMR1.CKS[2:0] bits are set to 000b, note that the characteristics of the digital noise filter may lead to the elimination of needed signals as noise.



Note: NFE: Digital noise filter circuit enable bit
NF[1:0]: Digital noise filter stage selection bits

Figure 26.23 Block Diagram of Digital Noise Filter Circuit

26.9 Address Match Detection

The RIIC can set three unique slave addresses in addition to the general call address and host address, and also can set 7- bit or 10-bit slave addresses.

26.9.1 Slave-Address Match Detection

The RIIC can set three unique slave addresses, and has a slave address detection function for each unique slave address. When the RIICnSER.SARy bit (y = 0 to 2) is set to 1, the slave addresses set in RIICnSARy (y = 0 to 2) can be detected. When the RIIC detects a match of the set slave address, the corresponding RIICnSR1.AASy flag (y = 0 to 2) is set to 1 at the rising edge of the ninth SCL clock cycle, and the RIICnSR2.RDRF flag or the RIICnSR2.TDRE flag is set to 1 by the following R/W# bit. This causes a receive data full interrupt (INTRIICRI) or transmit data empty interrupt (INTRIICTI) to be generated. The AASy flag is used to identify which slave address has been specified.

Figure 26.24 to **Figure 26.26** show the AASy flag set timing in three cases.

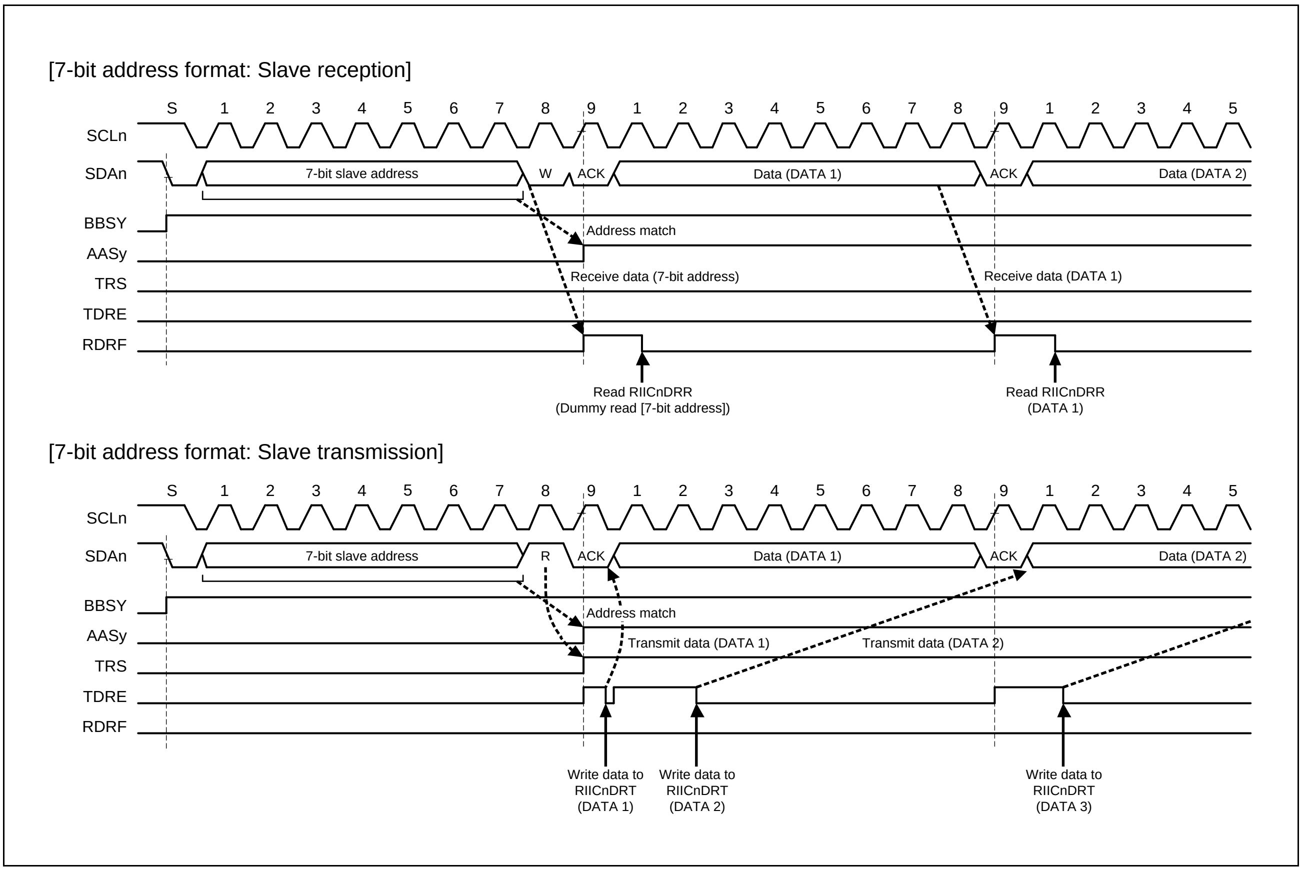


Figure 26.24 AASy Flag Set Timing with 7-Bit Address Format Selected

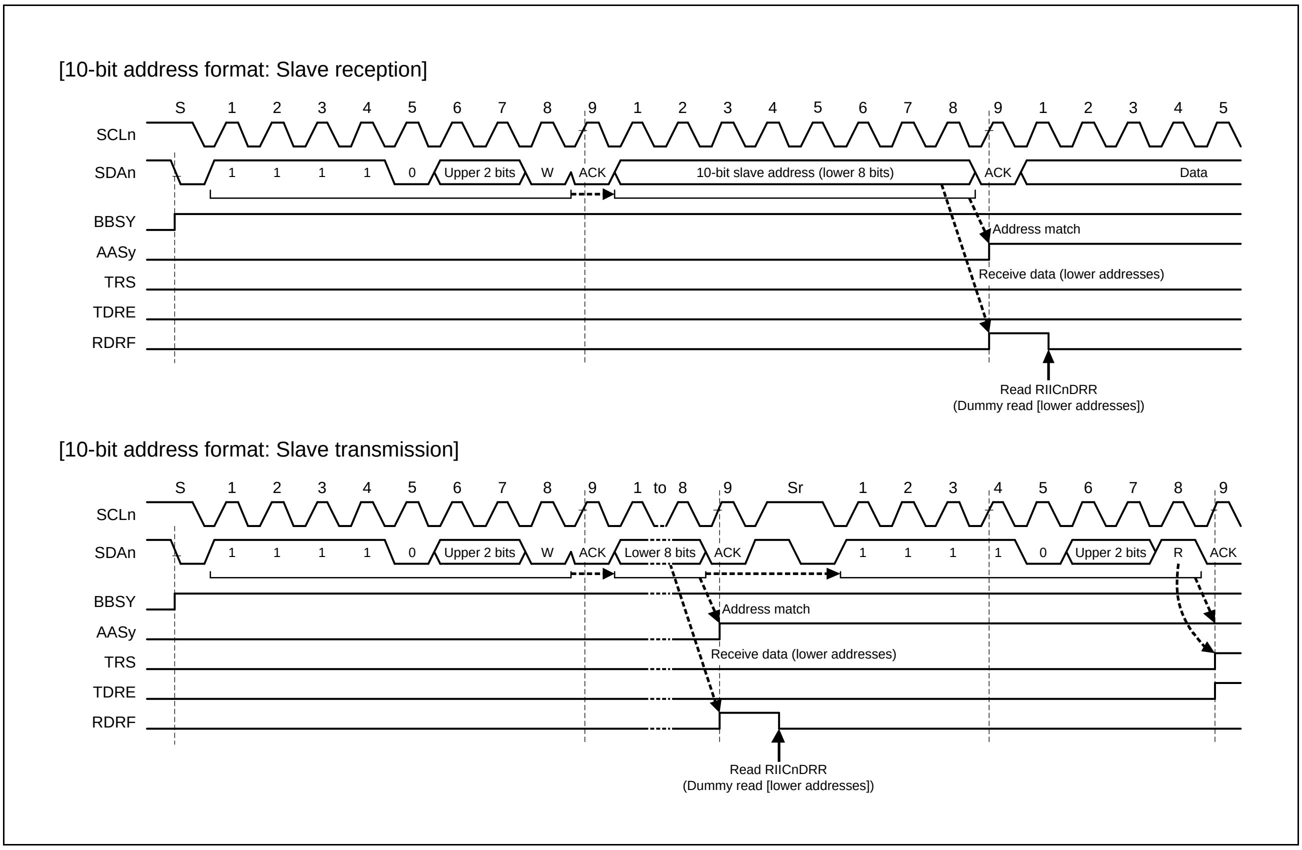


Figure 26.25 AASy Flag Set Timing with 10-Bit Address Format Selected

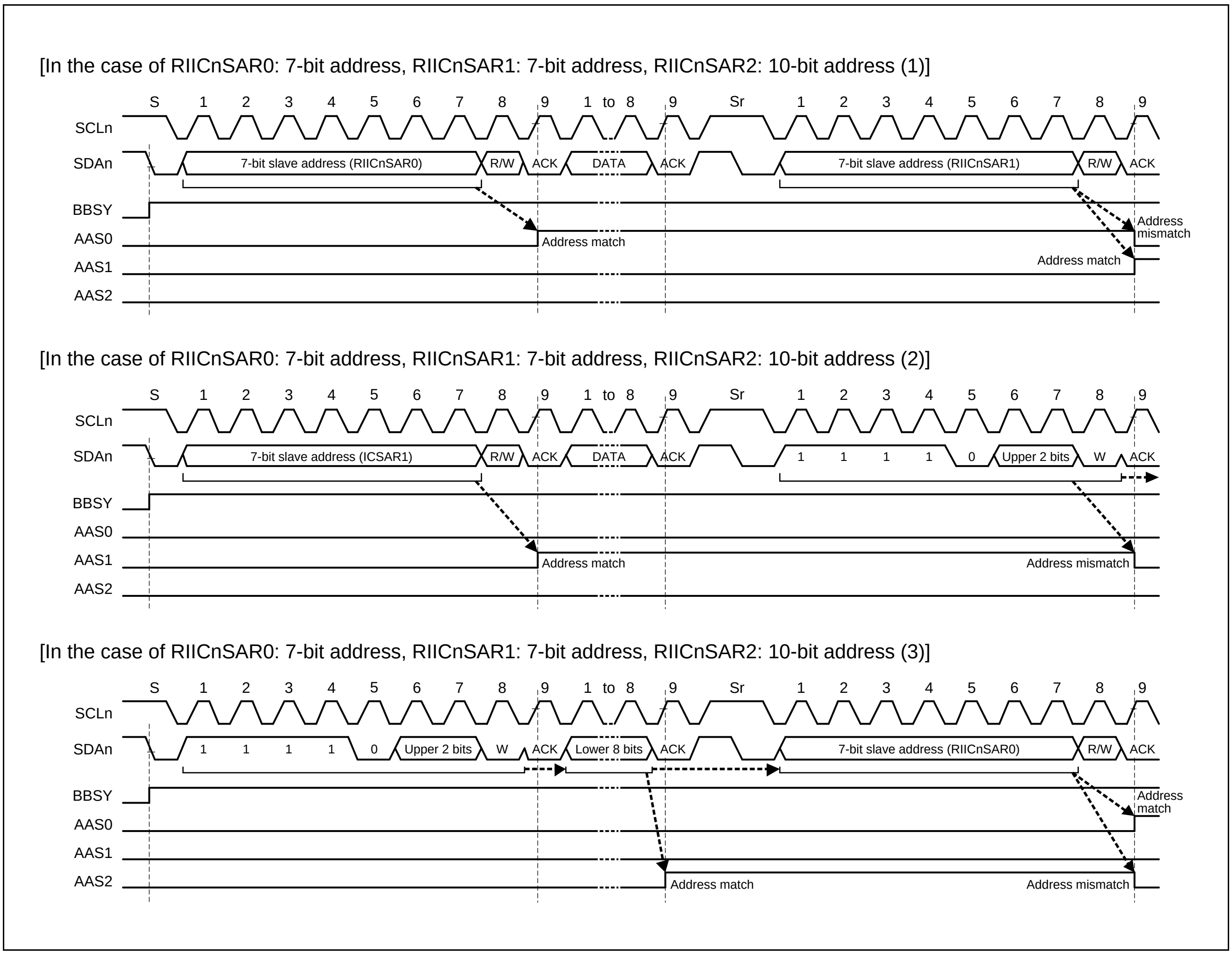


Figure 26.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed

26.9.2 Detection of the General Call Address

The RIIC has a facility for detecting the general call address (0000 000b + 0 [W]). This is enabled by setting the RIICnSER.GCE bit to 1.

If the address received after a start or restart condition is issued is 0000 000b + 1[R] (start byte), the RIIC recognizes this as the address of a slave device with an “all-zero” address but not as the general call address.

When the RIIC detects the general call address, both the RIICnSR1.GCA flag and the RIICnSR2.RDRF flag are set to 1 on the rising edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (INTRIICRI). The value of the GCA flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

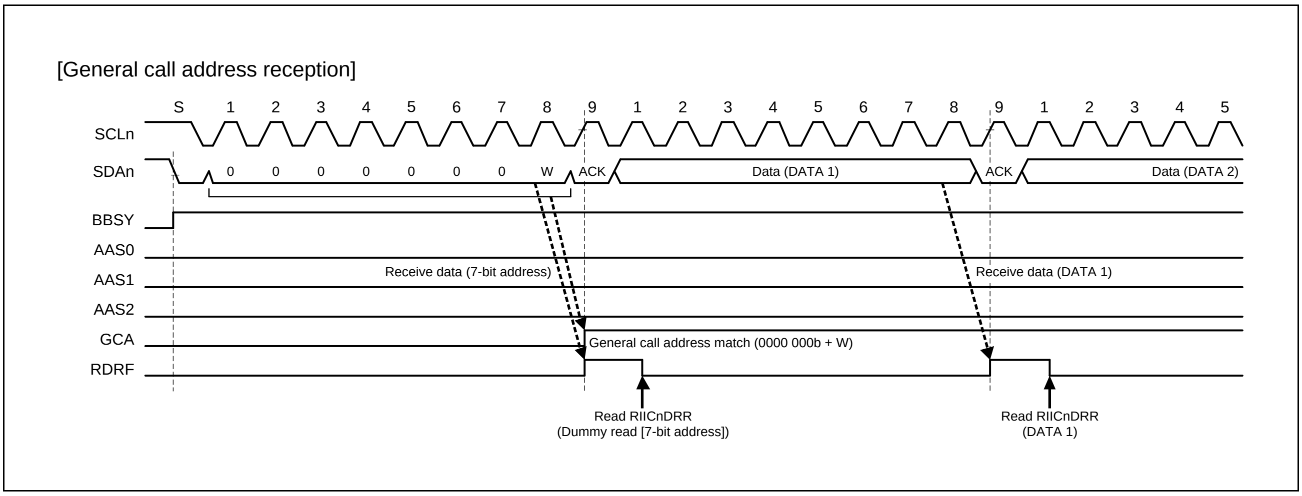


Figure 26.27 Timing of GCA Flag Setting during Reception of General Call Address

26.9.3 Device-ID Address Detection

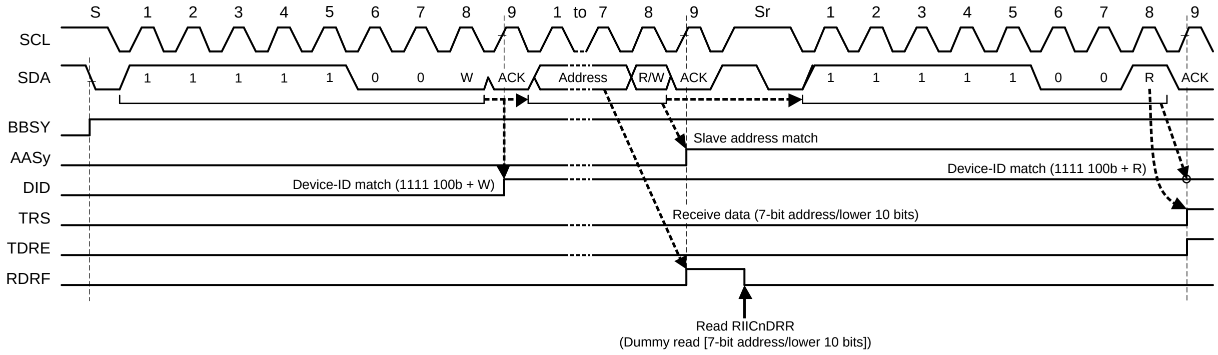
The RIIC module has a facility for detecting device-ID addresses conformant with the I^{2}C bus specification (Rev. 03). When the RIIC receives 1111 100b as the first byte after a start condition or restart condition was issued with the RIICnSER.DIDE bit set to 1, the RIIC recognizes the address as a device ID, sets the RIICnSR1.DID flag to 1 on the rising edge of the ninth SCL clock cycle when the following R/W# bit is 0, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, the RIIC sets the corresponding RIICnSR1.AASy flag (y = 0 to 2) to 1.

After that, when the first byte received after a start or restart condition is issued matches the device ID address (1111 100b) again and the following R/W# bit is 1, the RIIC does not compare the second and subsequent bytes and sets the RIICnSR2.TDRE flag to 1.

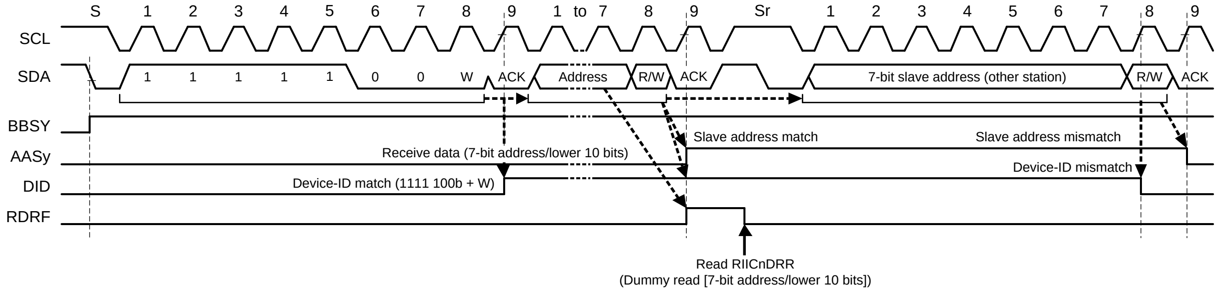
In the device-ID address detection function, the RIIC clears the DID flag to 0 if a match with the RIIC’s own slave address is not obtained or a match with the device ID address is not obtained after a match with the RIIC’s own slave address and the detection of a restart condition. If the first byte after detection of a start or restart condition matches the device ID address (1111 100b) and the R/W# bit is 0, the RIIC sets the DID flag to 1 and compares the second and subsequent bytes with the RIIC’s slave address. If the R/W# bit is 1, the DID flag holds the previous value and the RIIC does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DID flag after confirming that TDRE = 1.

Furthermore, prepare the device-ID fields (three bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details, see I^{2}C Bus Standard from NXP Semiconductors.

[Device-ID reception]



[When address received after a restart condition is detected does not match the Device-ID]



[When address before the Device-ID + R does not match the slave address]

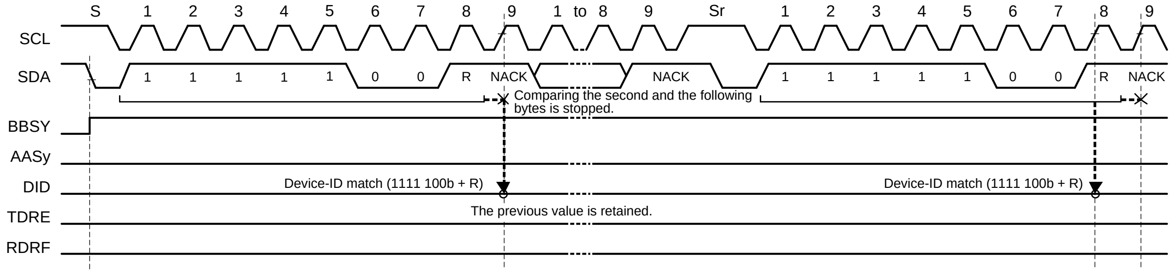


Figure 26.28 AASy/DID Flag Set/Clear Timing during Reception of Device-ID

26.9.4 Host Address Detection

The RIIC has a function to detect the host address while the SMBus is operating. When the RIICnSER.HOAE bit is set to 1 while the RIICnMR3.SMBS bit is 1, the RIIC can detect the host address (0001 000b) in slave receive mode (RIICnCR2.MST and TRS bits = 00b).

When the RIIC detects the host address, the RIICnSR1.HOA flag is set to 1 at the rising edge of the ninth SCL clock cycle, and at the same time, the RIICnSR2.RDRF flag is set to 1 when the R/W# bit is 0 (Wr bit). This causes a receive data full interrupt (INTRIICRI) to be generated. The HOA flag is used to recognize that the host address was sent from the smart battery or other devices.

If the bit following the host address (0001 000b) is an Rd bit (R/W# bit = 1), the RIIC can also detect the host address. After the host address is detected, the RIIC operates in the same manner as normal slave operation.

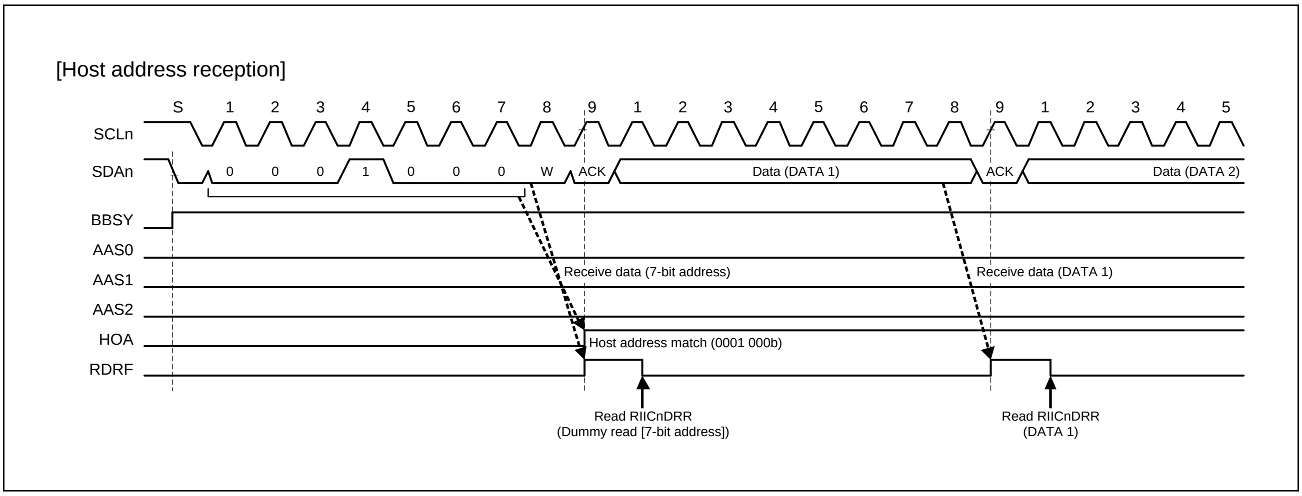


Figure 26.29 HOA Flag Set Timing during Reception of Host Address

26.10 Automatically Low-Hold Function for SCL

26.10.1 Function to Prevent Wrong Transmission of Transmit Data

If the shift register (RIICnDRS) is empty when data have not been written to the transmit data register (RIICnDRT) with the RIIC in transmission mode (RIICnCR2.TRS bit = 1), the SCL signal is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

<Master transmit mode>

- Low-level interval after a start condition or restart condition is issued
- Low-level interval of one clock cycle between the ninth clock cycle of one transfer and the first clock cycle of the next

<Slave transmit mode>

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

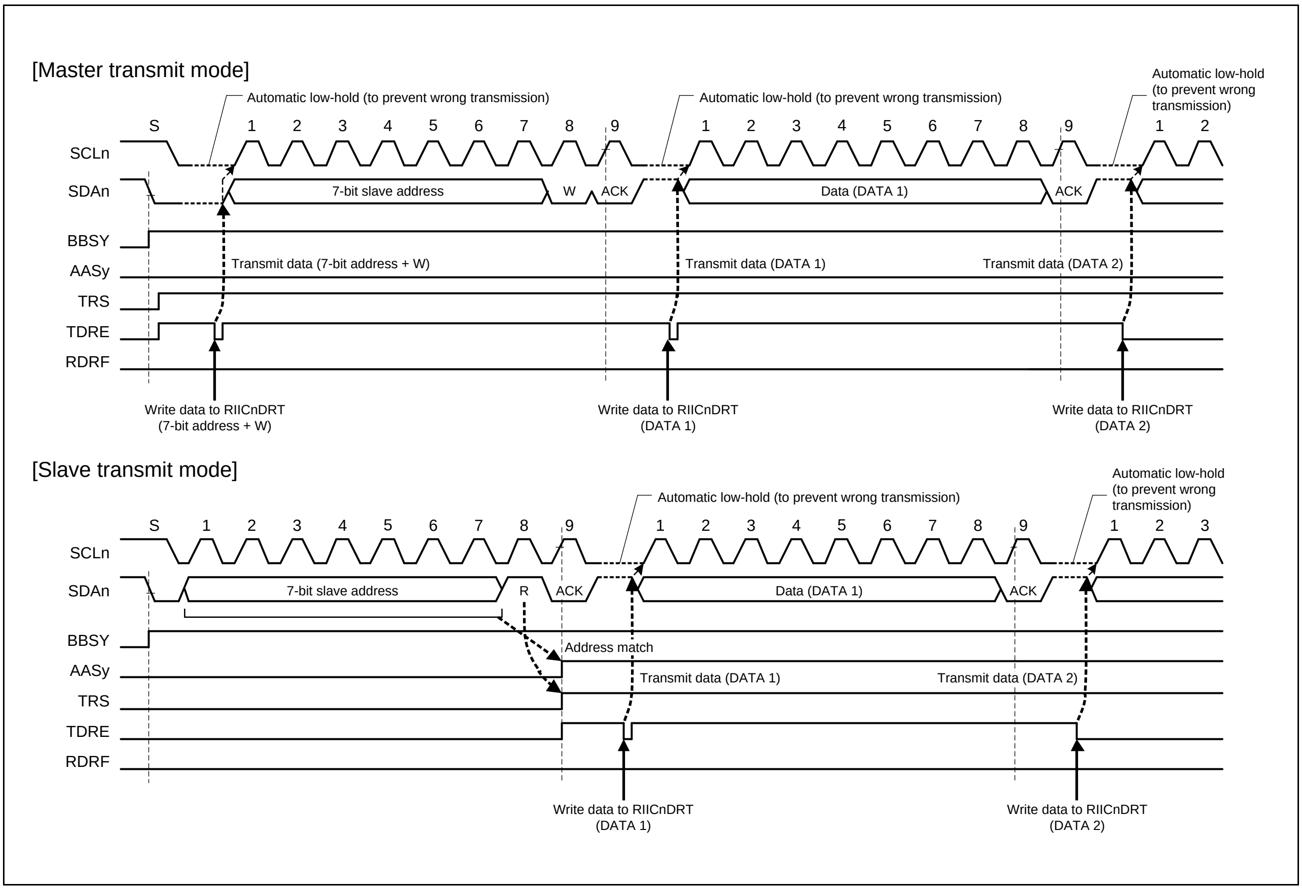


Figure 26.30 Automatic Low-Hold Operation in Transmit Mode

26.10.2 NACK Reception Transfer Suspension Function

The RIIC has a function to suspend transfer operation when NACK is received in transmit mode (RIICnCR2.TRS bit = 1). This function is enabled when the RIICnFER.NACKE bit is set to 1 (transfer suspension enabled). If the next transmit data has already been written (RIICnSR2.TDRE flag = 0) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically suspended. This prevents the SDA line output level from being held low when the MSB of the next transmit data is 0.

If the transfer operation is suspended by this function (RIICnSR2.NACKF flag = 1), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to clear the NACKF flag to 0. In master transmit mode, clear the NACKF flag to 0 after issuing a restart condition or clear the NACKF and STOP flags to 0 after confirming that a stop condition has been issued, and then issue a start condition.

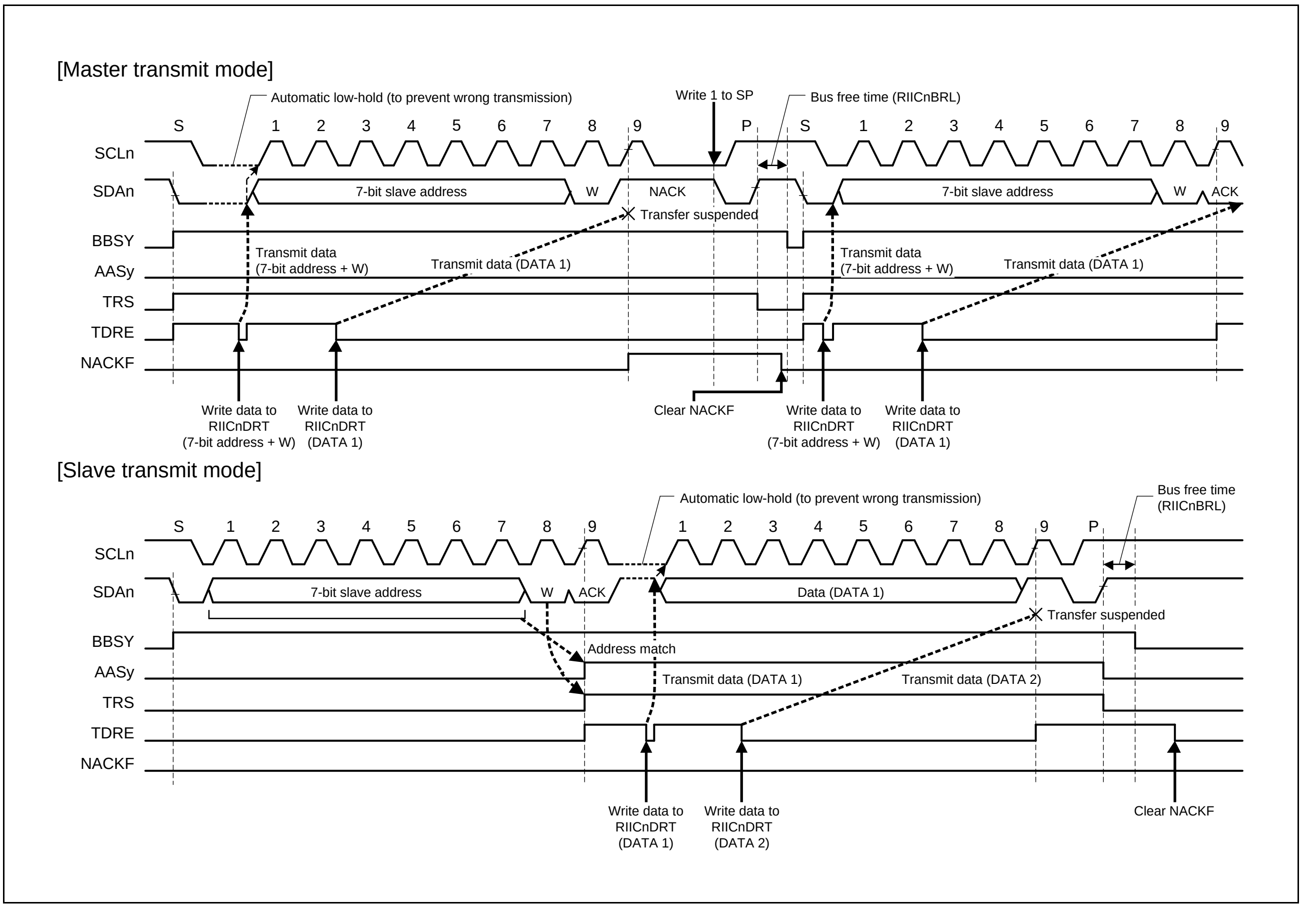


Figure 26.31 Suspension of Data Transfer when NACK is Received (NACKE = 1)

26.10.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (RIICnDRR) read is delayed for a period of one transfer frame or more with receive data full (RIICnSR2.RDRF flag = 1) in receive mode (RIICnCR2.TRS = 0), the RIIC holds the SCL line low automatically immediately before the next data is received to prevent failure to receive data.

This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC's own slave address is designated after a stop condition is issued. This function does not disturb other communication because the RIIC does not hold the SCL line low when a mismatch with its own slave address occurs after a stop condition is issued.

Sections in which the SCL line is held low can be selected with a combination of the RIICnMR3.WAIT and RDRFS bits.

(1) One-Byte Receive Operation and Automatic Low-Hold Function Using the WAIT Bit

When the RIICnMR3.WAIT bit is set to 1, the RIIC performs one-byte receive operation using the WAIT bit function. Furthermore, when the RIICnMR3.RDRFS bit is 0, the RIIC automatically sends the RIICnMR3.ACKBT bit value for the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle, and automatically holds the SCL line low at the falling edge of the ninth SCL clock cycle using the WAIT bit function. This low-hold is released by reading data from RIICnDRR, which enables bytewise receive operation.

The WAIT bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

(2) One-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the RDRFS Bit

When the RIICnMR3.RDRFS bit is set to 1, the RIIC performs one-byte receive operation using the RDRFS bit function. When the RIICnSR2.RDRFS bit is set to 1, the RDRF flag (receive data full) in RIICnSR2 is set to 1 at the rising edge of the eighth SCL clock cycle, and the SCL line is automatically held low at the falling edge of the eighth SCL clock cycle. This lowhold is released by writing a value to the RIICnMR3.ACKBT bit, but cannot be released by reading data from RIICnDRR, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.

The RDRFS bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

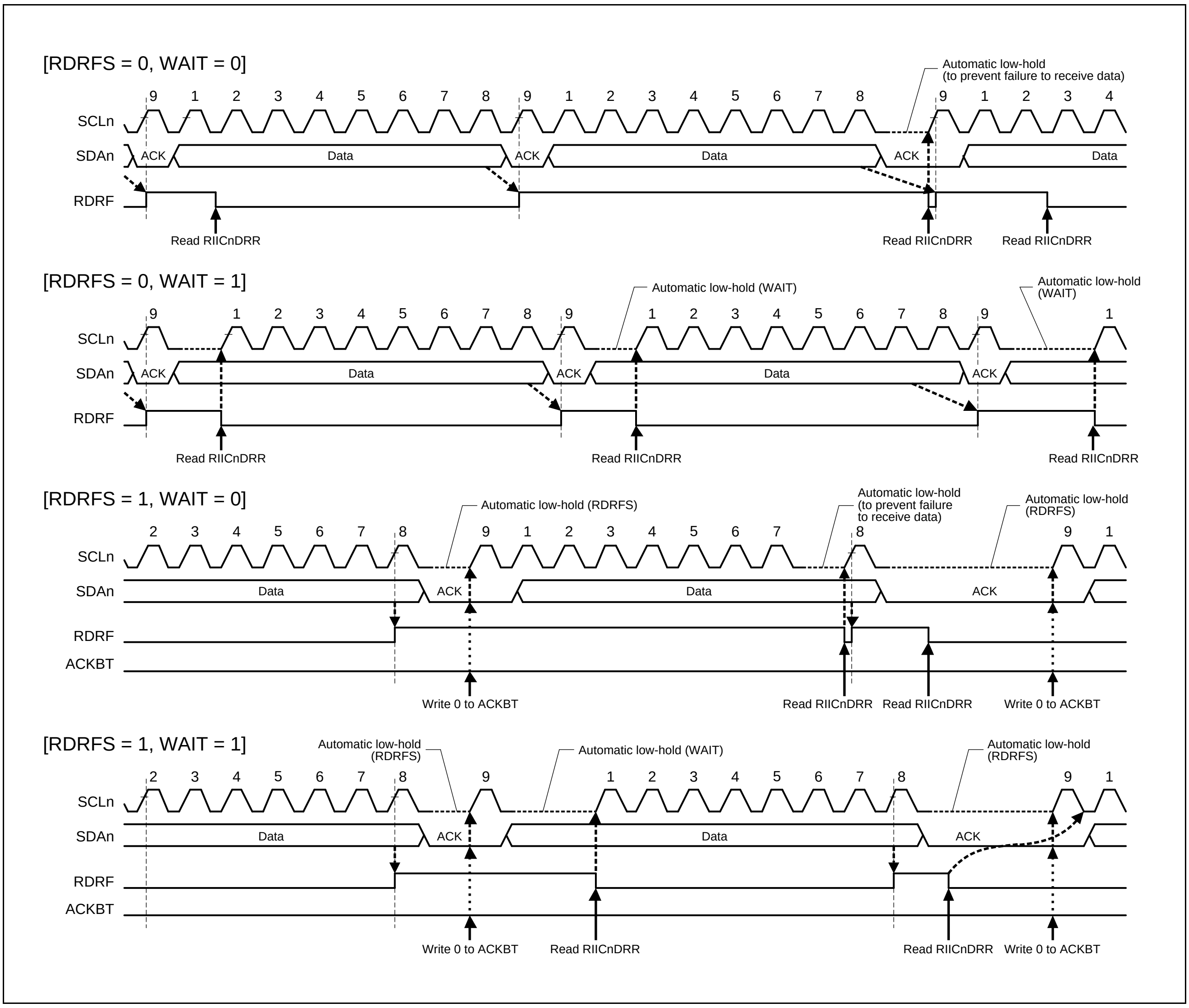


Figure 26.32 Automatic Low-Hold Operation in Receive Mode (Using RDRFS and WAIT Bits)

26.11 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I^{2}C bus standard, the RIIC has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration-lost in slave transmit mode.

26.11.1 Master Arbitration-Lost Detection (MALE Bit)

The RIIC drives the SDA line low to issue a start condition. However, if the SDA line has already been driven low by another master device issuing a start condition, the RIIC regards its own issuing of a start condition as an error and considers this a loss in arbitration, so priority is given to transfer by the other master device. Similarly, if a request to issue a start condition is made by setting the RIICnCR2.ST bit to 1 while the bus is busy (RIICnCR2.BBSY flag = 1), the RIIC regards this as a double-issuing-of-start-condition error and considers itself to have lost in arbitration, thus preventing a failure of transfer due to issuing of a start condition while transfer is in progress.

When a start condition is issued successfully, if the data for transmission including the address bits (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line, the RIIC loses in arbitration.

After a loss in arbitration of mastership, the RIIC immediately enters slave receive mode. If a slave address (including the general call address) matches its own address at this time, the RIIC continues in slave operation.

A loss in arbitration of mastership is detected when the following conditions are met while the RIICnFER.MALE bit is 1 (master arbitration-lost detection enabled).

[Master arbitration-lost conditions]

- Non-matching of the internal level for output on SDA and the level on the SDA line after a start condition was issued by setting the RIICnCR2.ST bit to 1 while the RIICnCR2.BBSY flag was cleared to 0 (erroneous issuing of a start condition)
- Setting of the RIICnCR2.ST bit to 1 (start condition double-issue error) while the RIICnCR2.BBSY flag is set to 1
- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDA line in master transmit mode (RIICnCR2.MST and TRS bits = 11b)

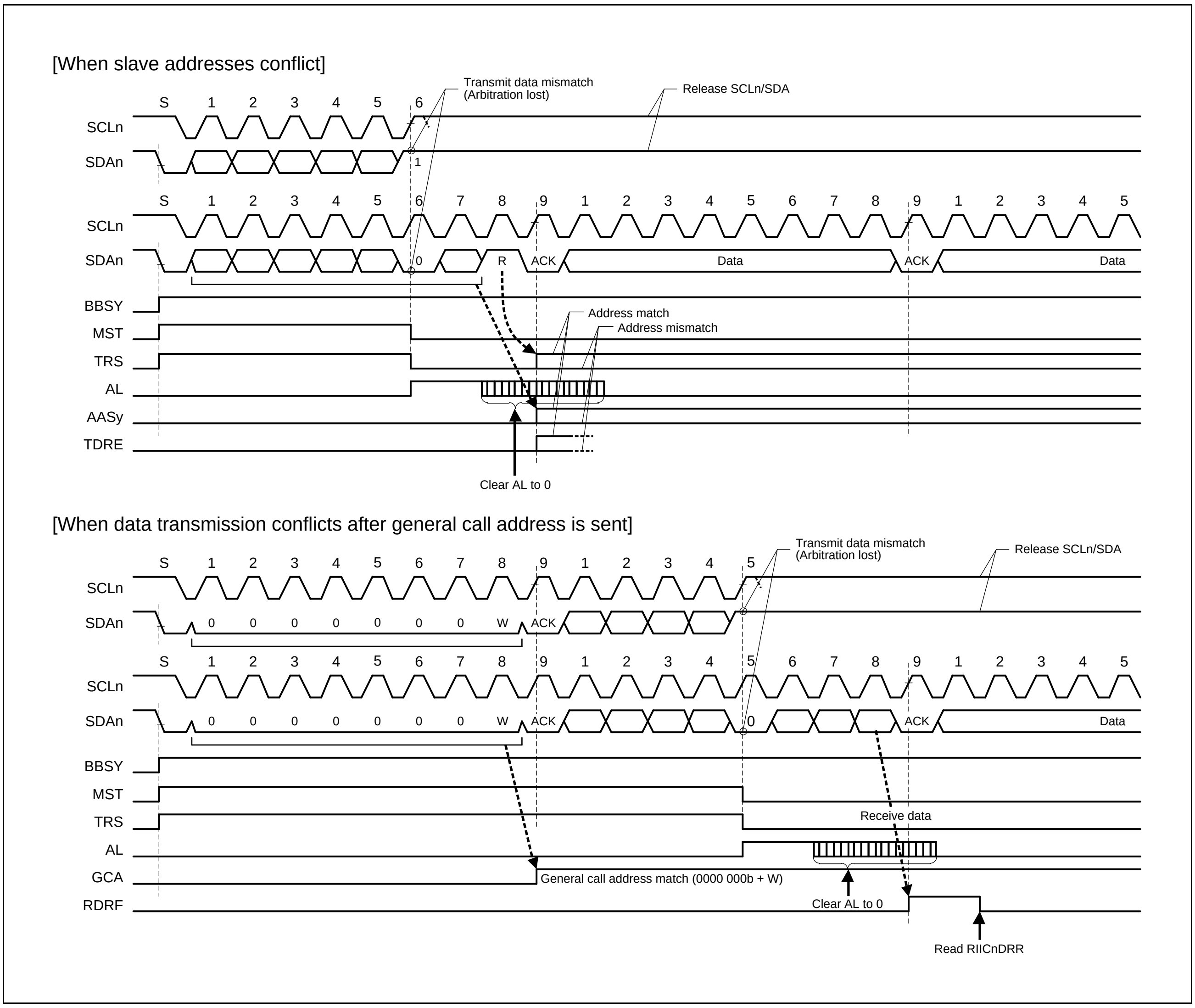


Figure 26.33 Examples of Master Arbitration-Lost Detection (MALE = 1)

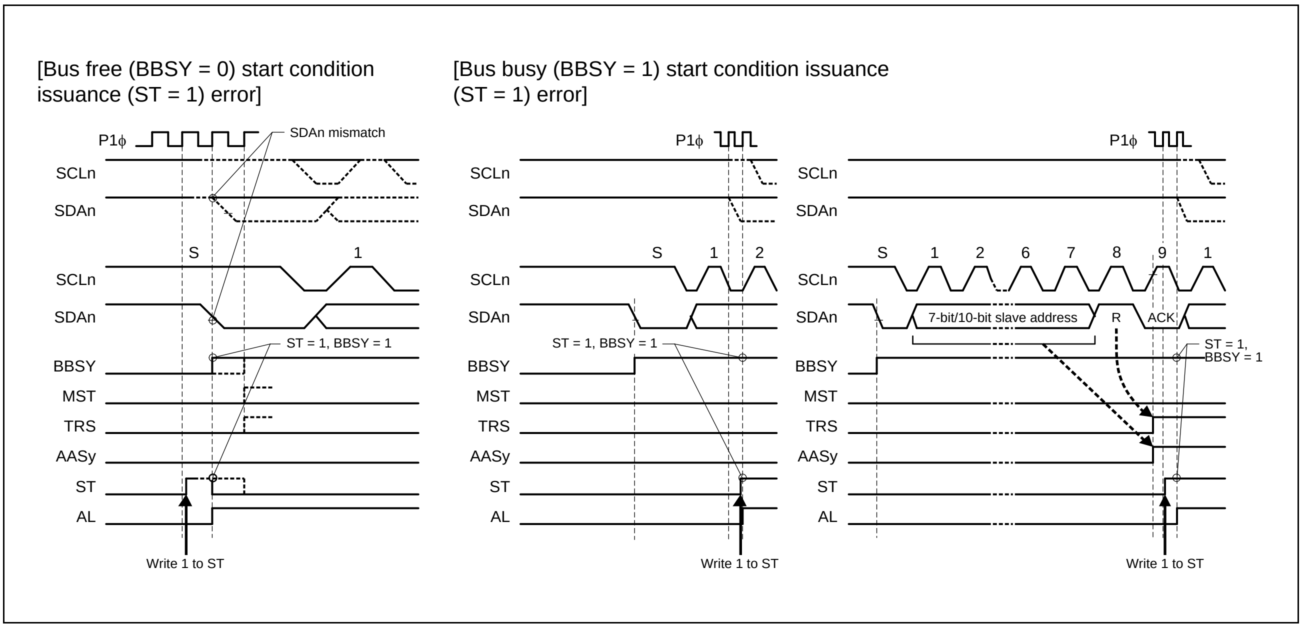


Figure 26.34 Arbitration-Lost when a Start Condition is Issued (MALE = 1)

26.11.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit)

The RIIC has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDA line (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line during transmission of NACK in receive mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device. **Figure 26.35** shows an example of arbitration-lost detection during transmission of NACK.

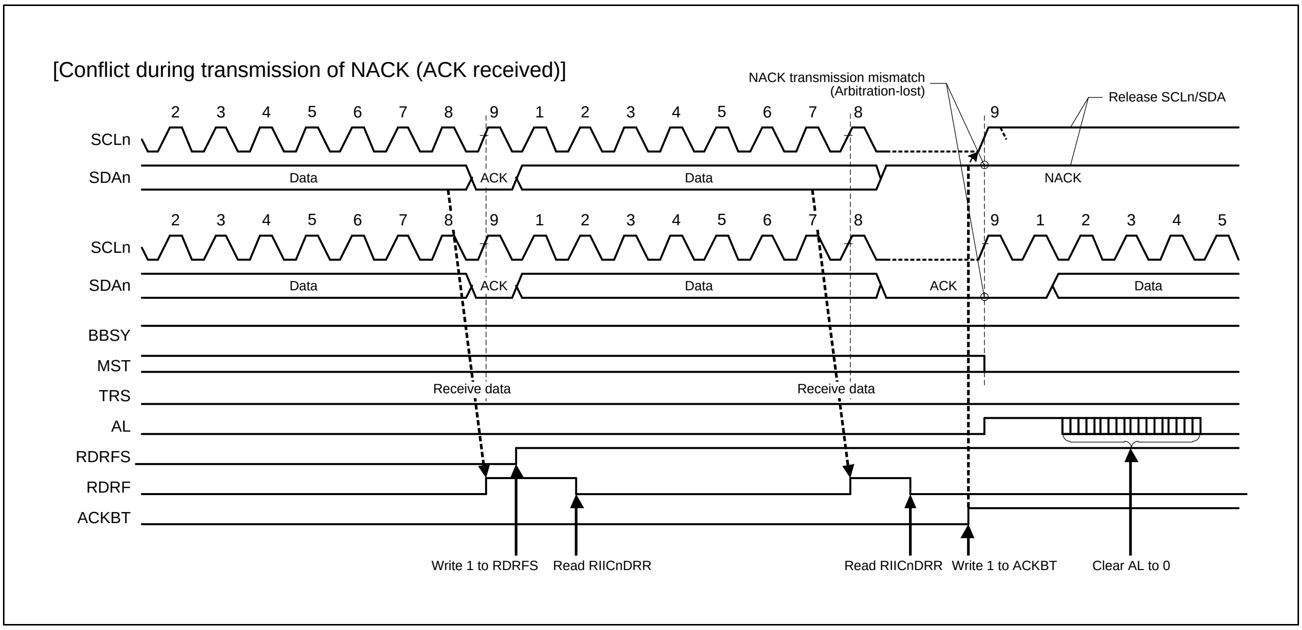


Figure 26.35 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1)

The following explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives two bytes of data from the slave device, and master B receives four bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Here, master A sends NACK when it has received two final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received necessary four bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition. Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When the RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, the RIIC enters slave receive mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

Similarly, in the ARP command processing of SMBus, the function to detect loss of arbitration during transmission of NACK is also available for eliminating the extra clock cycle processing (such as H’FF transmission processing) necessary if the UDID (Unique Device Identifier) of assign address does not match in the Get UDID (general) processing after the Assign address command.

The RIIC detects arbitration-lost during transmission of NACK when the following condition is met with the RIICnFER.NALE bit set to 1 (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

When the internal SDA output level does not match the SDA line (ACK is received) during transmission of NACK (RIICnMR3.ACKBT bit = 1).

26.11.3 Slave Arbitration-Lost Detection (SALE Bit)

The RIIC has a function to cause arbitration to be lost if the data for transmission (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the highimpedance state) and the low level is detected on the SDA line in slave transmit mode. This arbitration-lost detection function is mainly used when transmitting a UDID (Unique Device Identifier) over an SMBus.

When it loses slave arbitration, the RIIC enters slave receive mode. This function can detect conflicts of data during transmission of UDIDs over an SMBus and eliminates subsequent redundant processing (processing for the transmission of H’FF).

The RIIC detects slave arbitration-lost when the following condition is met with the RIICnFER.SALE bit set to 1 (slave arbitration-lost detection enabled). [Condition for slave arbitration-lost]

When transmit data excluding acknowledge (internal SDA output level) does not match the SDA line in slave transmit mode (RIICnCR2.MST and TRS bits = 01b)

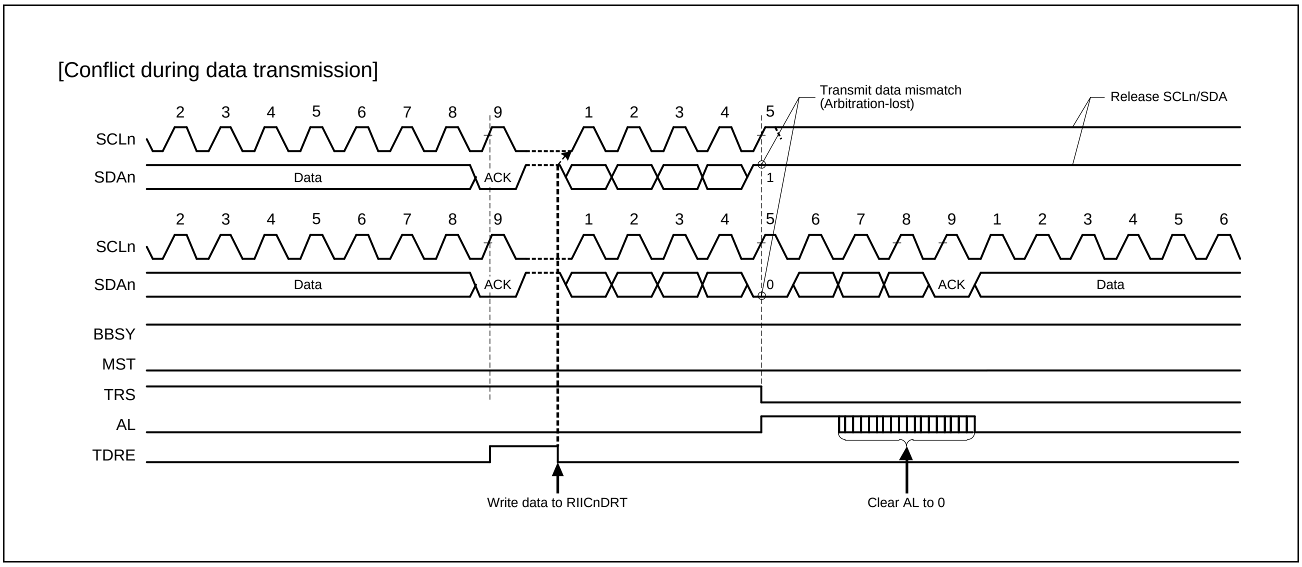


Figure 26.36 Example of Slave Arbitration-Lost Detection (SALE = 1)

26.12 Start Condition/Restart Condition/Stop Condition Issuing Function

26.12.1 Issuing a Start Condition

The RIIC issues a start condition when the RIICnCR2.ST bit is set to 1.

When the ST bit is set to 1, a start condition issuance request is made and the RIIC issues a start condition when the RIICnCR2.BBSY flag is 0 (bus free). When a start condition is issued normally, the RIIC automatically shifts to the master transmit mode.

A start condition is issued in the following sequence.

[Start condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the start condition hold time.
- Drive the SCL line low (high level to low level).
- Detect low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

26.12.2 Issuing a Restart Condition

The RIIC issues a restart condition when the RIICnCR2.RS bit is set to 1.

When the RS bit is set to 1, a restart condition issuance request is made and the RIIC issues a restart condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).

A restart condition is issued in the following sequence.

[Restart condition issuance]

- Release the SDA line.
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRL and the restart condition setup time.
- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the restart condition hold time.
- Drive the SCL line low (high level to low level).
- Detect a low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

CAUTION

To issue a restart condition request, set RIICnSR2.RS to 1 after setting RIICnSR2.START to 0. After checking that RIICnSR2.START is 1, set RIICnSR2.START to 0, and then write the slave address to RIICnDRT.

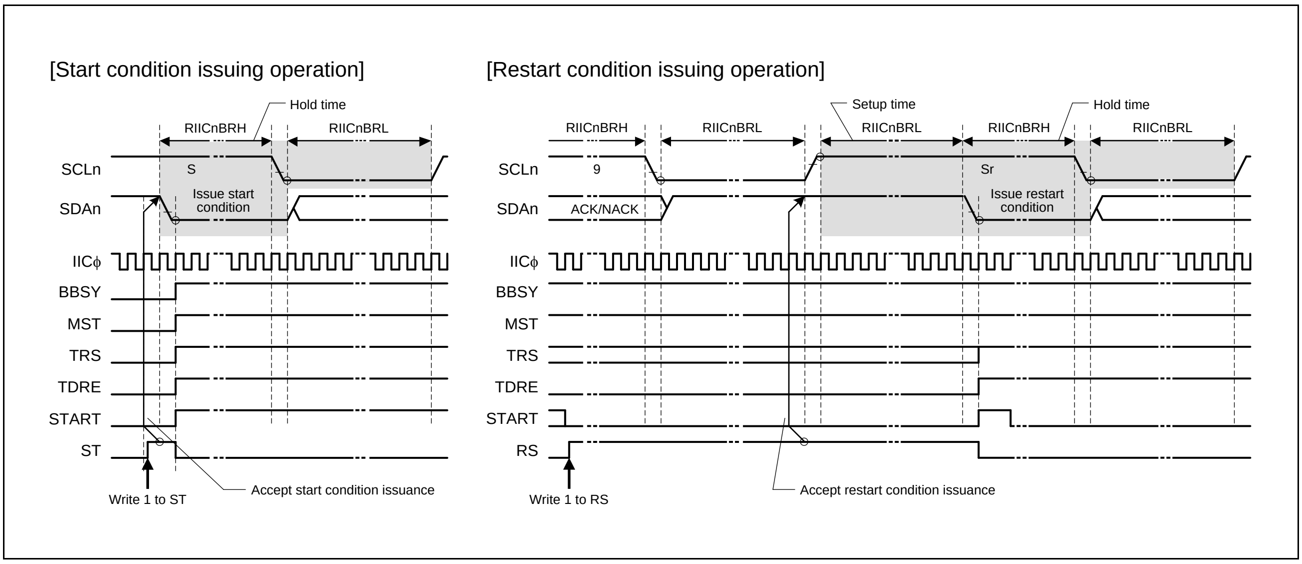


Figure 26.37 Start Condition/Restart Condition Issue Timing (ST and RS Bits)

26.12.3 Issuing a Stop Condition

The RIIC issues a stop condition when the RIICnCR2.SP bit is set to 1.

When the SP bit is set to 1, a stop condition issuance request is made and the RIIC issues a stop condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).

A stop condition is issued in the following sequence.

[Stop condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRH and the stop condition setup time.
- Release the SDA line (low level to high level).
- Ensure the time set in RIICnBRL and the bus free time.
- Clear the BBSY flag to 0 (to release the bus mastership).

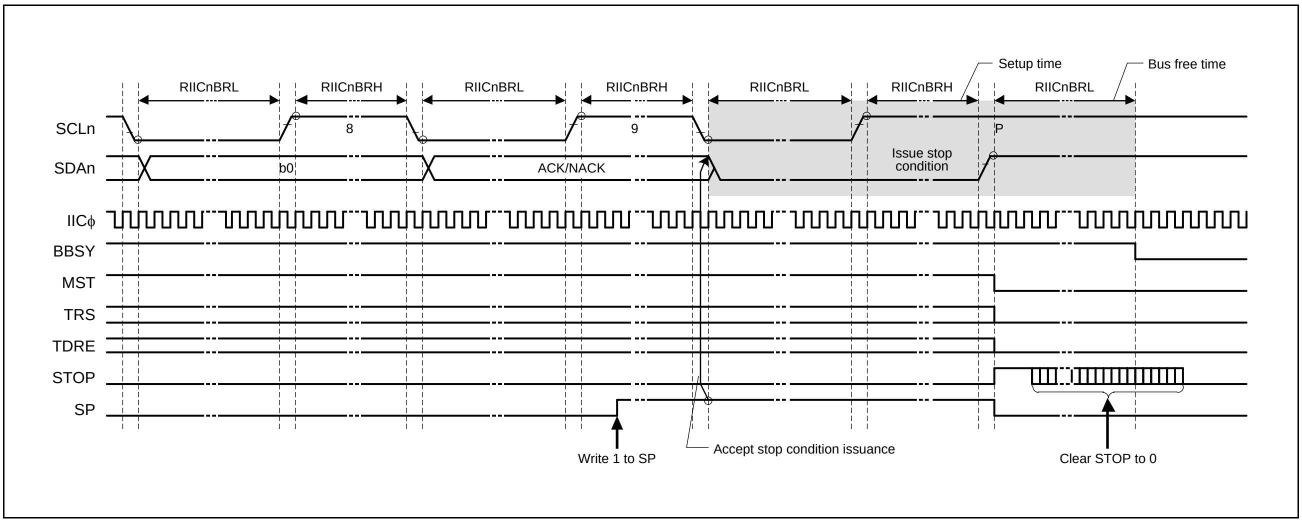


Figure 26.38 Stop Condition Issue Timing (SP Bit)

26.13 Bus Hanging

If the clock signals from the master and slave devices go out of synchronization due to noise or other factors, the I^{2}C bus might hang with a fixed level on the SCL line and/or SDA line.

As measures against the bus hanging, the RIIC has a timeout function to detect hanging by monitoring the SCL line, a function for the output of an extra SCL clock cycle to release the bus from a hung state due to clock signals being out of synchronization, and the RIIC/internal reset function.

By checking the RIICnCR1.SCLO, SDAO, SCLI, and SDAI bits, it is possible to see whether the RIIC or its partner in communications is placing the low level on the SCL or SDA lines.

26.13.1 Timeout Function

The RIIC has the timeout function to detect an abnormality that the SCL line is held for a certain period of time. The RIIC can detect an abnormal bus state by monitoring that the SCL line is held low or high for a predetermined time. The timeout function monitors the SCL line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCL line changes (rising or falling), but continues to count unless the SCL line changes. If the internal counter overflows due to no SCL line change, the RIIC can detect the timeout and report the bus abnormality.

This timeout function is enabled when the RIICnFER.TMOE bit is 1. It detects an abnormal bus state in which the SCL line is held low or high in the following cases.

1. When the bus is busy (RIICnCR2.BBSY = 1) in master mode (RIICnCR2.MST = 1)
2. When the bus is busy (RIICnCR2.BBSY = 1) and the RIIC's own slave address matches (RIICnSR1 is not H'00) in slave mode (RIICnCR2.MST = 0)
3. While the bus is free (RIICnCR2.BBSY = 0) and issuing of a start condition is being requested (RIICnCR2.ST = 1).

The internal counter of the timeout function works using the internal reference clock (IICϕ) set by the RIICnMR1.CKS[2:0] bits as a count source. It functions as a 16-bit counter when long mode is selected (RIICnMR2.TMOS bit = 0) or a 14-bit counter when short mode is selected (TMOS bit = 1).

The SCL line level (low/high or both levels) during which this counter is activated can be selected by the setting of the RIICnMR2.TMOH and TMOL bits. If both TMOL and TMOH bits are cleared to 0, the internal counter does not work.

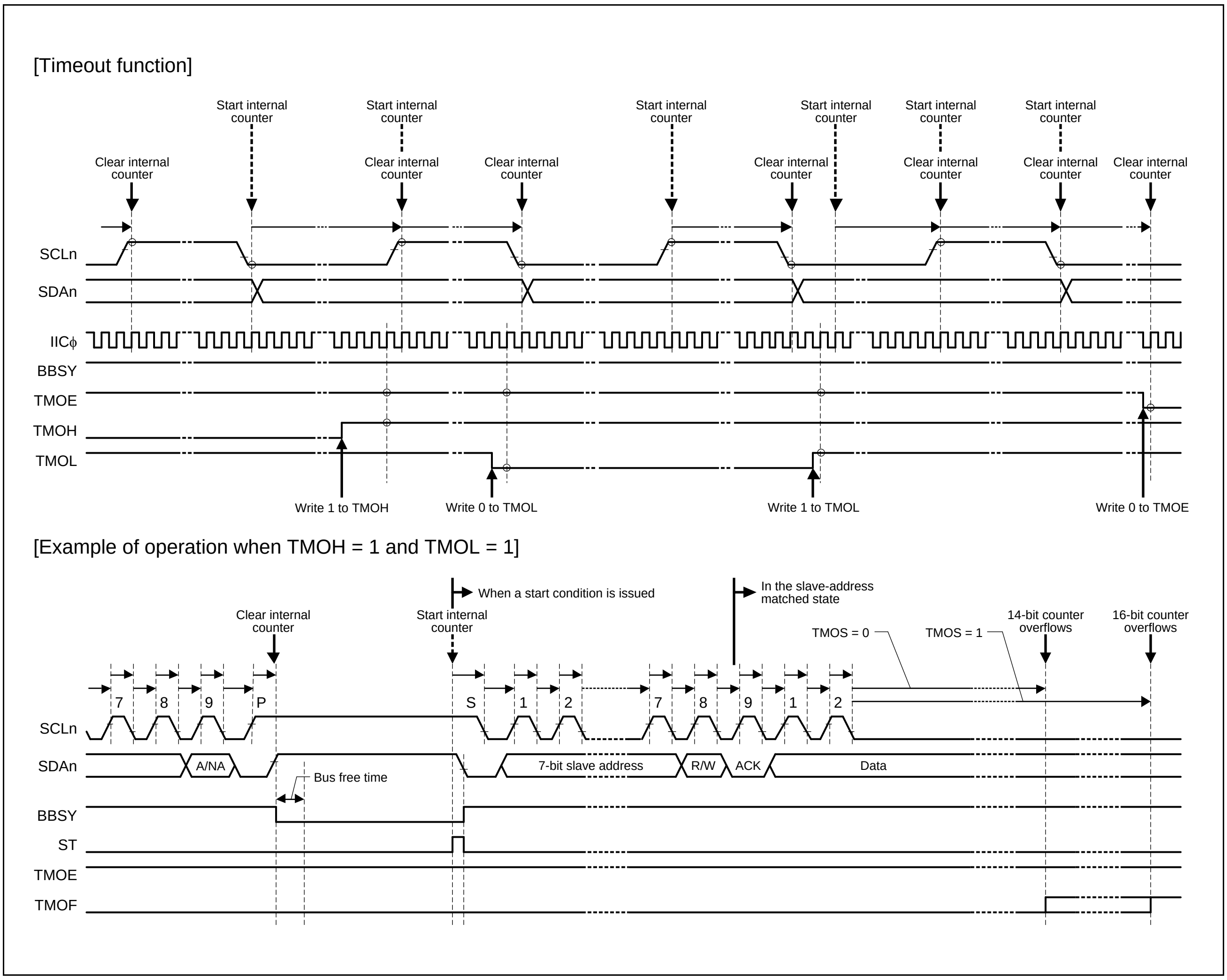


Figure 26.39 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits)

26.13.2 Extra SCL Clock Cycle Output Function

In master mode, the RIIC module has a facility for the output of extra SCL (clock) cycles to release the SDA line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.

This function is mainly used in master mode to release the SDA line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from the RIIC with single cycles of the SCL (clock) signal as the unit in the case of a bus error where the RIIC cannot issue a stop condition because the slave device is holding the SDA line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.

When the RIICnCR1.CLO bit is set to 1 in master mode, a single cycle of the SCL clock at the frequency corresponding to the transfer rate settings (settings of the RIICnMR1.CKS[2:0] bits, and of the RIICnBRH and RIICnBRL registers) is output as an extra clock cycle. After output of this single cycle of the SCL clock, the CLO bit is automatically cleared to 0. At this point, the SCL pin output is held at a low level if the BBSY flag is 1, or the SCL pin outputs at high level if the BBSY flag is 0. Further clock cycles can consecutively be output by writing 1 to the CLO bit after having read CLO is 0 by software.

When the RIIC module is in master mode and the slave device is holding the SDA line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a stop condition is not possible. The facility for output of an extra cycle of the SCL (clock) signal can be used to output extra cycles of SCL one by one to make the slave device release the SDA line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDA line by the slave device can be monitored by reading the RIICnCR1.SDAI bit. After confirming release of the SDA line by the slave device, complete communications by reissuing the stop condition.

[Output conditions for using the RIICnCR1.CLO bit]

- When the bus is free (RIICnCR2.BBSY flag = 0) or in master mode (RIICnCR2.MST bit = 1 and BBSY flag = 1)
- When the communication device does not hold the SCL line low

Figure 26.40 shows the operation timing of the extra SCL clock cycle output function (CLO bit).

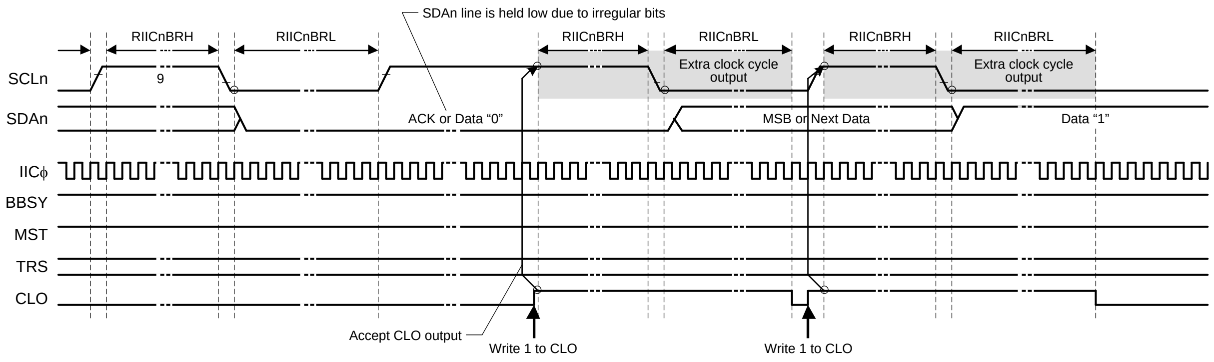


Figure 26.40 Extra SCL Clock Cycle Output Function (CLO Bit)

26.13.3 RIIC Reset and Internal Reset

The RIIC module incorporates a function for resetting itself. There are two types of reset. One is referred to as an RIIC reset; this initializes all registers including the RIICnCR2.BBSY flag. The other is referred to as an internal reset; this releases the RIIC from the slave-address matched state and initializes the internal counter while retaining other settings. After issuing a reset, be sure to clear the RIICnCR1.IICRST bit to 0.

Both types of reset are effective for release from bus-hung states since both restore the output state of the SCL and SDA pins to the high impedance state.

Issuing a reset during slave operation may lead to a loss of synchronization between the master device clock and the slave device clock, so avoided this where possible. Note that monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (RIICnCR1.ICE and IICRST bits = 01b).

For a detailed description of the RIIC and internal resets, see **Section 26.15, Reset Function of RIIC**.

26.14 SMBus Operation

The RIIC is available for data communication conforming to the SMBus (Version 2.0). To perform SMBus communication, set the RIICnMR3.SMBS bit to 1. To use the transfer rate within a range of 10 kbps to 100 kbps of the SMBus standard, set the RIICnMR1.CKS[2:0] bits, RIICnCBRH, and RIICnBRL. In addition, determine the values of the RIICnMR2.DLCS bit and the RIICnMR2.SDDL[2:0] bits to meet the data hold time specification of 300 ns or more. If the RIIC is used only as a slave device, the transfer rate setting is not necessary. When the RIIC is used only as a slave device, the transfer rate setting is not necessary, whereas the RIICnBRL needs to be set to a value at least the same as the data setup time (250 ns).

For the SMBus device default address (1100 001b), use one of the slave address registers L0 to L2 (RIICnSAR0, RIICnSAR1, and RIICnSAR2), and set the corresponding RIICnSARy.FSy bit (7-bit/10-bit address format select) (y = 0 to 2) to 0 (7-bit address format).

When transmitting the UDID (Unique Device Identifier), set the RIICnFER.SALE bit to 1 to enable the slave arbitration lost detection function.

26.14.1 SMBus Timeout Measurement

(1) Measuring timeout of slave device

The following period (timeout interval: $T_{LOW:\,SEXT}$) must be measured for slave devices in SMBus communication.

- From start condition to stop condition

  To measure timeout for slave devices, measure the period from start condition detection to stop condition detection with the internal timer using a start condition detection interrupt (INTRIICSTI) and stop condition detection interrupt (INTRIICSPI) of the RIIC. The measured timeout period must be within the total clock low-level period [slave device] $T_{LOW:\,SEXT}$: 25 ms (max.) of the SMBus standard.

  If the time measured with the internal timer exceeds the clock low-level detection timeout TTIMEOUT: 25 ms (min.) of the SMBus standard, the slave device must release the bus by writing 1 to the RIICnCR1.IICRST bit to issue an internal reset of the RIIC. When an internal reset is issued, the RIIC stops driving the bus for the SCL pin and SDA pin and make the SCL/SDA pin outputs high impedance, which releases the bus.

(2) Measuring timeout of master device

The following periods (timeout interval: $T_{LOW:\,MEXT}$) must be measured for master devices in SMBus communication.

- From start condition to acknowledge bit
- Between acknowledge bits
- From acknowledge bit to stop condition

To measure timeout for master devices, measure these periods with the internal timer using a start condition detection interrupt (INTRIICSTI), stop condition detection interrupt (INTRIICSPI), and transmit end interrupt (INTRIICTEI) or receive data full interrupt (INTRIICRI) of the RIIC. The measured timeout period must be within the total clock low-level extended period [master device] $T_{LOW:\,MEXT}$: 10 ms (max.) of the SMBus standard, and the total of all $T_{LOW:MEXT}$ from start condition to stop condition must be within $T_{LOW:\,SEXT}$: 25 ms (max.).

For the ACK receive timing (rising edge of the ninth SMBCLK clock cycle), monitor the RIICnSR2.TEND flag in master transmit mode (master transmitter) and the RIICnSR2.RDRF flag in master receive mode (master receiver). For this reason, perform bytewise transmit operation in master transmit mode, and hold the RIICnMR3.RDRFS bit 0 until the byte just before reception of the final byte in master receive mode. While the RDRFS bit is 0, the RDRF flag is set to 1 at the rising edge of the ninth SMBCLK clock cycle.

If the period measured with the internal timer exceeds the total clock low-level extended period [master device] $T_{LOW:MEXT}$: 10 ms (max.) of the SMBus standard or the total of measured periods exceeds the clock low-level detection timeout TTIMEOUT: 25 ms (min.) of the SMBus standard, the master device must stop the transaction by issuing a stop condition. In master transmit mode, immediately stop the transmit operation (writing data to RIICnDRT).

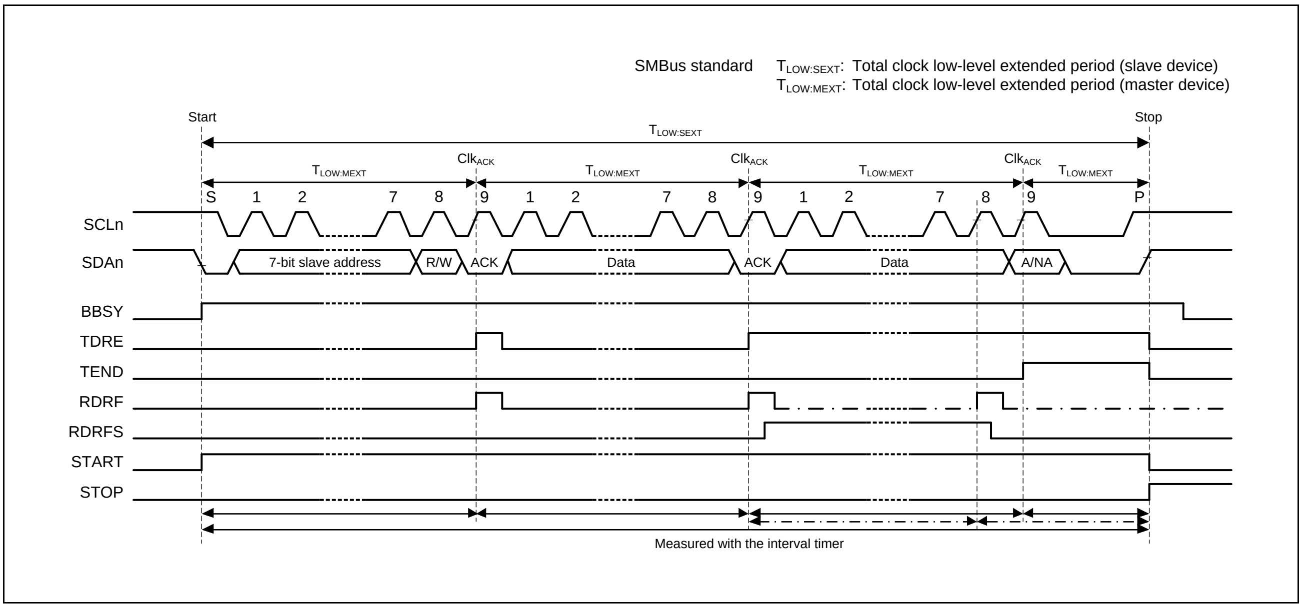


Figure 26.41 SMBus Timeout Measurement

26.14.2 SMBus Host Notification Protocol/Notify ARP Master

In communications over an SMBus, a slave device can temporarily act as a master device to notify the SMBus host (or ARP master) of (or request the SMBus host for) its own slave address or to request its own slave address from the SMBus host.

For this LSI to operate as an SMBus host (or ARP master), the host address (0001 000b) sent from the slave device must be detected as a slave address, so the RIIC has a function for detecting the host address. To detect the host address as a slave address, set the RIICnMR3.SMBS bit and the RIICnSER.HOAE bit to 1. Operation after the host address has been detected is the same as normal slave operation.

26.15 Reset Function of RIIC

The RIIC has chip reset, RIIC reset, and internal reset functions. **Table 26.12** lists the scope of each reset and reset conditions.

Table 26.12 RIIC Reset Functions (1/2)

Register		RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/Restart Condition Detection	Stop Condition Detection
RIICnCR1	ICE	0	1	Retained	Retained
	IICRST	1	1	Retained	Retained
	CLO	Initialized	Retained	Retained	Retained
	SOWP	Initialized	Retained	Retained	Retained
	SCLO	Initialized	Initialized	Retained	Retained
	SDAO	Initialized	Initialized	Retained	Retained
	SCLI	Initialized	Retained	Retained	Retained
	SDAI	Initialized	Retained	Retained	Retained
RIICnCR2	BBSY	Initialized	Initialized*1	Operation	Retained
	MST	Initialized	Initialized	Operation (retained)	Initialized
	TRS	Initialized	Initialized	Operation (retained)	Initialized
	SP	Initialized	Initialized	Initialized	Initialized
	RS	Initialized	Initialized	Initialized	Initialized
	ST	Initialized	Initialized	Initialized	Retained
RIICnMR1	CKS[2:0]	Initialized	Retained	Retained	Retained
	BCWP	Initialized	Retained	Retained	Retained
	BC[2:0]	Initialized	Initialized	Initialized	Retained
RIICnMR2		Initialized	Retained	Retained	Retained
RIICnMR3	WAIT	Initialized	Retained	Retained	Retained
	RDRFS	Initialized	Retained	Retained	Retained
	ACKWP	Initialized	Retained	Retained	Retained
	ACKBT	Initialized	Retained	Retained	Initialized
	ACKBR	Initialized	Retained	Retained	Retained
	NF[1:0]	Initialized	Retained	Retained	Retained
RIICnFER		Initialized	Retained	Retained	Retained
RIICnSER		Initialized	Retained	Retained	Retained
RIICnIER		Initialized	Retained	Retained	Retained
RIICnSR1	DID	Initialized	Initialized	Retained	Initialized
	GCA	Initialized	Initialized	Retained	Initialized
	AAS2	Initialized	Initialized	Retained	Initialized
	AAS1	Initialized	Initialized	Retained	Initialized
	AAS0	Initialized	Initialized	Retained	Initialized

Table 26.12 RIIC Reset Functions (2/2)

Register		RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/Restart Condition Detection	Stop Condition Detection
RIICnSR2	TDRE	Initialized	Initialized	Retained	Initialized
	TEND	Initialized	Initialized	Retained	Initialized
	RDRF	Initialized	Initialized	Retained	Retained
	NACKF	Initialized	Initialized	Retained	Retained
	STOP	Initialized	Initialized	Retained	Operation
	START	Initialized	Initialized	Operation	Initialized
	AL	Initialized	Initialized	Retained	Retained
	TMOF	Initialized	Initialized	Retained	Retained
RIICnSAR0, RIICnSAR1, RIICnSAR2		Initialized	Retained	Retained	Retained
RIICnBRH, RIICnBRL		Initialized	Retained	Retained	Retained
RIICnDRT		Initialized	Retained	Retained	Retained
RIICnDRR		Initialized	Retained	Retained	Retained
RIICnDRS		Initialized	Initialized	Retained	Retained

Note 1. When an internal reset is applied while the bus is free after detection of a stop condition, the setting of the BBSY flag is 0 while the bus is free following de-assertion of the internal reset signal.
When an internal reset is applied while the bus is not free, the BBSY flag is not cleared.

27. Serial Sound Interface (SSIF-2)

27.1 Overview

The serial sound interface (SSIF-2) transmits and receives audio data to and from various devices that are compatible with I^2S format, monaural format and TDM format.

27.1.1 Features

Table 27.1 Features of SSIF-2

Item		Description
Number of interfaces		- 4 (n = 0, 1, 2, 3) (RZ/G2L) - 3 (n = 0, 1, 2) (RZ/G2LC)
Communication mode		- Master/slave - Full-duplex communication is available in interfaces 0, 1, and 3. Half-duplex communication is only available in interface 2.
Communication format		- I^2S format - Monaural format - TDM format
Serial data		- MSB first - Data can be left-justified or right-justified. - Data delay (one clock cycle) or no delay selectable for the period from SSI_RCK to SSI_TXD/SSI_RXD - System word length: 8, 16, 24, 32, 48, 64, 128, or 256 bits - Data word length: 8, 16, 18, 20, 22, 24, or 32 bits - Padding polarity: Low or high
Bit clock (SSIBCK)	In master mode	- Clock source division ratio: 1/1, 1/2, 1/4, 1/6, 1/8, 1/12, 1/16, 1/24, 1/32, 1/48, 1/64, 1/96, and 1/128. - Supply/stop of SSI_BCK is selectable while communication is halted.
	In slave mode	- SSI_BCK input method:Direct input or via the noise canceller
	In master/slave mode	- Polarity: Rising/falling edge canceler
Left and right clock/frame synchronization (SSILRCK/SSIFS)	In master mode	- Polarity: Low/high level selectable - Supply/stop of SSI_RCK is selectable while communication is halted.
	In slave mode	- Two input methods: Direct input or via the noise canceller
Transmit data (SSITxD) and receive data (SSIRxD)	Transmission	- Mute: Transmission of transmit FIFO data or fixed value 0 selectable
	Reception (in slave mode)	- Two input methods: Direct input or via the noise canceller
FIFO	Capacity	- Transmit FIFO/receive FIFO: 4 bytes × 32 stages, - Receive FIFO: 4 bytes × 32 stages
	Data alignment	- Method of data transfer between FIFO and shift register (Left-justification and right-justification) selectable.
Interrupt	Interrupt output	- Communication error/idle mode error (level) - Receive data full interrupt (edge) - Transmit data empty interrupt (edge) - Receive data full/transmit data empty interrupt (edge)
	Interrupt capture spilling solution function	- Edge Interrupt supports inhibition function of capture spilling.
Low power consumption function		- Supply/stop of audio clock is selectable in master-mode communication.

The definition of words and phrases used by the format of the communication of SSIF-2 is as follows.

Table 27.2 Word Definition List

Word	Definition
Start trigger	The first edge that reaches value that set to LRCKP in SSI_RCK pin after SSIF-2 permit communication.
Frame boundary	When SSIF-2 begins to transmit the first data of one frame. When SSIF-2 forwarding ends final data of one frame.
Frame word number	Sound channel number in one frame
System word length	Bit length in one frame.
Data word length	Effective bit length in one frame.
Control bit of communication format	● SSICR register: FRM,DWL,SWL,LRCKP,SPDP,SDTA,PDTA,DEL ● SSIFCR register: BSW ● SSIOFR register: OMOD ● SSISCR register: TDES, RDFS

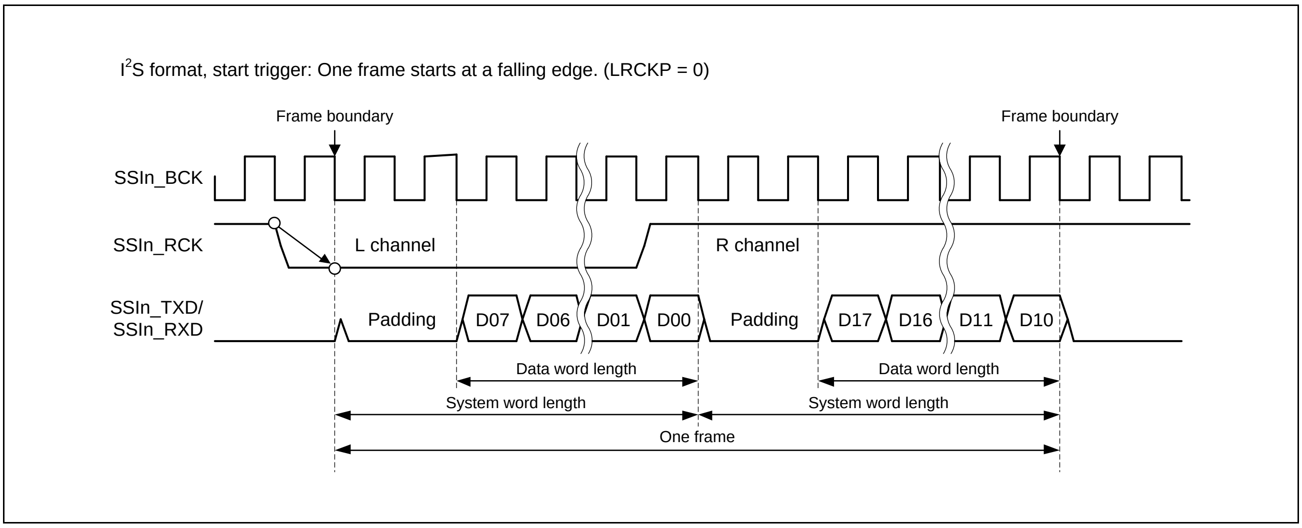
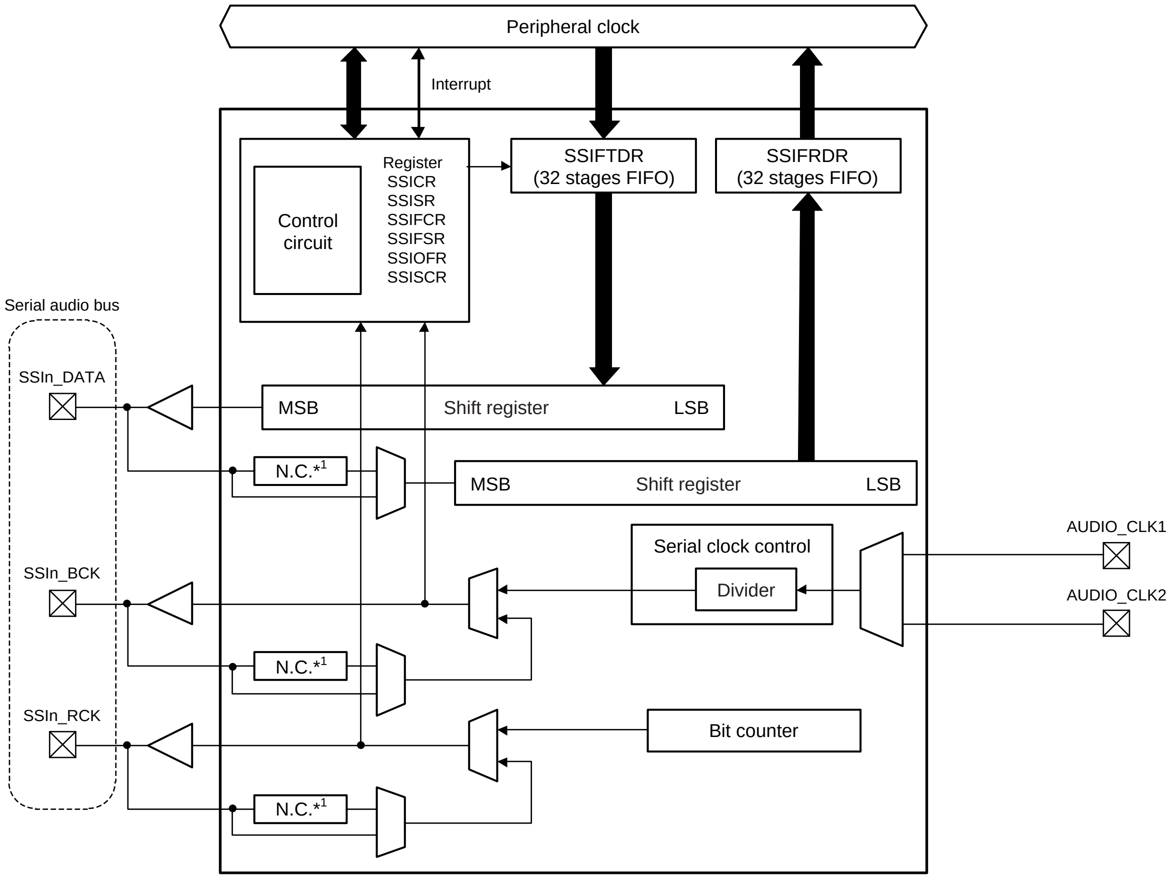


Figure 27.1 Communication Format Definition

27.1.2 Block Diagram

Figure 27.2 shows the block diagram of SSIF-2.



[Legend]

SSICR: Control register

SSISR: Status register

SSIFCR: FIFO control register

SSIFSR: FIFO status register

SSIOFR: Audio format register

SSIFTDR: Transmit FIFO data register

SSIFRDR: Receive FIFO data register

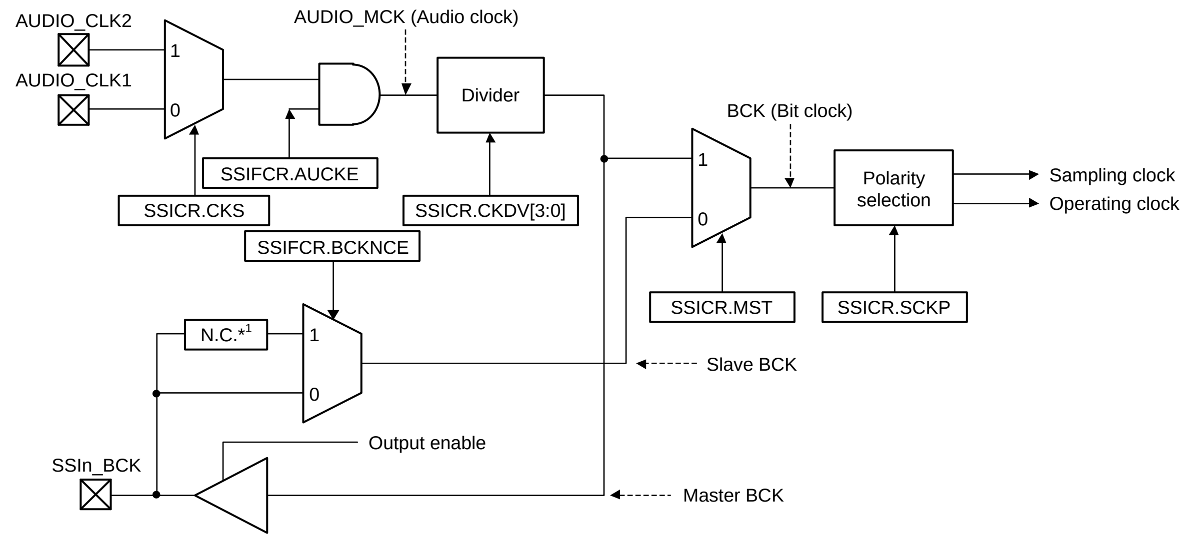
SSISCR: Status Control register

Note: In channels 0, 1, and 3, SSn_DATA can be used independently as SSIn_TXD for transmission and SSIn_RXD for reception.

Note 1. N.C.: Noise canceller

Figure 27.2 SSIF-2 Block Diagram

Figure 27.3 shows the clock configuration of SSIF-2.



[Legend]

SSICR: Control register

Note: Switch the above SFRs while communication is halted.

Note 1. N.C. (Noise canceller)

Figure 27.3 SSIF-2 Clock Configuration

27.2 Input/Output Pins

Table 27.3 shows the pin configuration

Table 27.3 Pin Configuration

Interface	name	I/O	Function
0, 1, 3 (n = 0,1,3)	SSIn_BCK	I/O	Bit clock
	SSIn_RCK	I/O	LR clock/frame synchronization
	SSIn_TXD	Output	Serial data output
	SSIn_RXD	Input	Serial data input
2 (n = 2)	SSIn_BCK	I/O	Bit clock
	SSIn_RCK	I/O	LR clock/frame synchronization
	SSIn_DATA	I/O	Serial data I/O
Common	AUDIO_CLK1	Input	External clock1 for audio (The oversampling clock signal is input.)
	AUDIO_CLK2	Input	External clock2 for audio (The oversampling clock signal is input.)

27.3 List of Registers

Table 27.4 shows the register configuration of SSIF-2. The address of the SSIF-2 register is represented by the offset address from the base address. SSIF-2 base address is as follows:

SSIF-2 base address: H’0_1004_9C00 (ch0), H’0_1004_A000 (ch1), H’0_1004_A400 (ch2), H’0_1004_A800 (ch3) (Overall Address Space)

SSIF-2 base address: H’5004_9C00 (ch0), H’5004_A000 (ch1), H’5004_A400 (ch2), H’5004_A800 (ch3) (Cortex-M33 Address Space Secure)

SSIF-2 base address: H’4004_9C00 (ch0), H’4004_A000 (ch1), H’4004_A400 (ch2), H’4004_A800 (ch3) (Cortex-M33 Address Space Non-Secure)

Table 27.4 List of Control Registers (1/2)

Channel	Register Name	Symbol	R/W	Initial Value	Address	Access Size
0	Control register	SSICR	RW	H’0000_0000	H’00	32 bits
	Status register	SSISR	RW	H’0200_0000	H’04	32 bits
	Note: Bits 31, 30, and 25 to 0 are read only.					
	FIFO control register	SSIFCR	RW	H’0000_0000	H’10	32 bits
	FIFO status register	SSIFSR	RW	H’0001_0000	H’14	32 bits
	Note: Bits 31 to 17 and 15 to 1 are read only.					
	Transmit FIFO data register	SSIFTDR	W	H’0000_0000	H’18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H’0000_0000	H’1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H’0000_0000	H’20	32 bits
	Status control register	SSISCR	RW	H’0000_0000	H’24	32 bits
1	Control register	SSICR	RW	H’0000_0000	H’00	32 bits
	Status register	SSISR	RW	H’0200_0000	H’04	32 bits
	Note: Bits 31, 30, and 25 to 0 are read only.					
	FIFO control register	SSIFCR	RW	H’0000_0000	H’10	32 bits
	FIFO status register	SSIFSR	RW	H’0001_0000	H’14	32 bits
	Note: Bits 31 to 17 and 15 to 1 are read only.					
	Transmit FIFO data register	SSIFTDR	W	H’0000_0000	H’18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H’0000_0000	H’1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H’0000_0000	H’20	32 bits
	Status control register	SSISCR	RW	H’0000_0000	H’24	32 bits

Table 27.4 List of Control Registers (2/2)

Channel	Register Name	Symbol	R/W	Initial Value	Address	Access Size
2	Control register	SSICR	RW	H'0000_0000	H'00	32 bits
	Status register	SSISR	RW	H'0200_0000	H'04	32 bits
	Note: Bits 31, 30, and 25 to 0 are read only.					
	FIFO control register	SSIFCR	RW	H'0000_0000	H'10	32 bits
	FIFO status register	SSIFSR	RW	H'0001_0000	H'14	32 bits
	Note: Bits 31 to 17 and 15 to 1 are read only.					
	Transmit FIFO data register	SSIFTDR	W	H'0000_0000	H'18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H'0000_0000	H'1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H'0000_0000	H'20	32 bits
	Status control register	SSISCR	RW	H'0000_0000	H'24	32 bits
3	Control register	SSICR	RW	H'0000_0000	H'00	32 bits
	Status register	SSISR	RW	H'0200_0000	H'04	32 bits
	Note: Bits 31, 30, and 25 to 0 are read only.					
	FIFO control register	SSIFCR	RW	H'0000_0000	H'10	32 bits
	FIFO status register	SSIFSR	RW	H'0001_0000	H'14	32 bits
	Note: Bits 31 to 17 and 15 to 1 are read only.					
	Transmit FIFO data register	SSIFTDR	W	H'0000_0000	H'18	32 bits, 16 bits, 8 bits
	Receive FIFO data register	SSIFRDR	R	H'0000_0000	H'1C	32 bits, 16 bits, 8 bits
	Audio format register	SSIOFR	RW	H'0000_0000	H'20	32 bits
	Status control register	SSISCR	RW	H'0000_0000	H'24	32 bits

Note: Access to the addresses other than those listed above is prohibited.

27.4 Function Details

27.4.1 Register Descriptions

27.4.1.1 Control Register (SSICR)

This is a 32-bit readable/writable register. With this register, select an audio clock, control IRQ, select data formats, and set an operation mode.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved. Write 0. The read value is 0.
30	CKS	0	RW	Selects an audio clock for master-mode communication*1 0: Selects the AUDIO_CLK2 input 1: Selects the AUDIO_CLK1 input
29	TUIEN	0	RW	Transmit underflow interrupt output enable 0: Disables transmit underflow interrupt output 1: Enables transmit underflow interrupt output
28	TOIEN	0	RW	Transmit overflow interrupt output enable 0: Disables transmit overflow interrupt output 1: Enables transmit overflow interrupt output
27	RUIEN	0	RW	Receive underflow interrupt output enable 0: Disables receive underflow interrupt output 1: Enables receive underflow interrupt output
26	ROIEN	0	RW	Receive overflow interrupt output enable 0: Disables receive overflow interrupt output 1: Enables receive overflow interrupt output
25	IIEN	0	RW	Idle mode interrupt output enable 0: Disables idle mode interrupt output 1: Enables idle mode interrupt output
24	—	0	R	Reserved. Write 0. The read value is 0.
23, 22	FRM[1:0]	00b	RW	Select frame word number*1

Communication format (SSIOFR.OMOD[1:0])

FRM[1:0]	I^2S (00b)	Monaural (10b)	TDM (01b)
00b	2	1	Set prohibition
01b	Set prohibition	Set prohibition	4
10b			6
11b			8

Bit	Bit Name	Initial Value	R/W	Description
21 to 19	DWL[2:0]	000b	R/W	Selects data word length*[1] 000: 8 bits 001: 16 bits 010: 18 bits 011: 20 bits 100: 22 bits 101: 24 bits 110: 32 bits 111: Setting prohibited
18 to 16	SWL[2:0]	000b	RW	Selects system word length*[1] 000: 8 bits 001: 16 bits 010: 24 bits 011: 32 bits 100: 48 bits 101: 64 bits 110: 128 bits 111: 256 bits
15	—	0	R	Reserved. Write 0. The read value is 0.
14	MST	0	RW	Master enable*[1] 0: Slave-mode communication 1: Master-mode communication
13	BCKP	0	RW	Selects bit clock polarity*[1] 0: SSILRCK/SSIFS and SSIFTxD/SSIFRxD change at a falling edge (sampling SSILRCK/SSIFS and SSIFRxD at a rising edge of SSIBCK). 1: SSILRCK/SSIFS and SSIFTxD/SSIFRxD change at a rising edge (sampling SSILRCK/SSIFS and SSIFRxD at a falling edge of SSIBCK).
12	LRCKP	0	RW	Selects the initial value and polarity of LRCK/FS*[1] 0: The initial value is at a high level For SSILRCK/SSIFS, one frame start trigger is falling edge of SSILRCK/SSIFS. 1: The initial value is at a low level For SSILRCK/SSIFS, one frame start trigger is rising edge of SSILRCK/SSIFS.
11	SPDP	0	RW	Selects serial padding polarity*[1] 0: Padding data is at a low level 1: Padding data is at a high level
10	SDTA	0	RW	Selects serial data alignment*[1] 0: Transmits and receives serial data first and then padding bits. 1: Transmit and receives padding bits first and then serial data.
9	PDTA	0	RW	Selects placement data alignment*[1] 0: Left-justifies placement data (SSIFTDR, SSIFRDR) 1: Right-justifies placement data (SSIFTDR, SSIFRDR)
8	DEL	0	RW	Selects serial data delay*[1] 0: Delay of one cycle of SSIBCK between SSILRCK/SSIFS and SSIFTxD/SSIFRxD. 1: No delay between SSILRCK/SSIFS and SSIFTxD/SSIFRxD. This bit control SSILRCK/SSIFS in monaural format (Refer to **Section 27.4.2.2, Monaural Format** for details).

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	CKDV[3:0]	H'0	RW	Selects bit clock division ratio*[1] CKDV[3:0] 0000: AUDIO_MCK 0001: AUDIO_MCK/2 0010: AUDIO_MCK/4 0011: AUDIO_MCK/8 0100: AUDIO_MCK/16 0101: AUDIO_MCK/32 0110: AUDIO_MCK/64 0111: AUDIO_MCK/128 1000: AUDIO_MCK/6 1001: AUDIO_MCK/12 1010: AUDIO_MCK/24 1011: AUDIO_MCK/48 1100: AUDIO_MCK/96 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
3	MUEN	0	RW	Mute (silent) enable 0: Disables muting on the next frame boundary. 1: Enables muting on the next frame boundary.
2	—	0	R	Reserved. Write 0. The read value is 0.
1	TEN	0	RW	Transmission and reception enable*[2]
0	REN	0	RW	(TEN, REN): Operating 00: Disables transmission and reception 01: Enables reception (starts reception) 10: Enables transmission (starts transmission) 11: Enables transmission and reception (starts transmission and reception)

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Note 2. If the TEN bit or REN bit is rewritten, make sure that the SSISR.IIRQ bit is in the desired status. If written, the operation performed immediately after writing is not guaranteed. For example, after enabling operation, make sure that SSISR.IIRQ = 0 and after disabling operation, make sure that SSISR.IIRQ = 1.

CKS Bit

This bit sets the audio clock in master-mode communication (MST = 1). In slave-mode communication (MST = 0), setting of this bit is invalid.

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 27.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

TUIEN Bit

This bit enables/disables output of transmit underflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.TUIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.TUIRQ = 1.

TOIEN Bit

This bit enables/disables output of transmit overflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.TOIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.TOIRQ = 1.

RUIEN Bit

This bit enables/disables output of receive underflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.RUIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.RUIRQ = 1.

ROIEN Bit

This bit enables/disables output of receive overflow interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.ROIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.ROIRQ = 1.

IIEN Bit

This bit enables/disables output of idle mode interrupts. By enabling this bit (set it to 1), an interrupt is output at a rising edge of SSISR.IIRQ = 1. An interrupt is also output when this bit is changed from 0 to 1 while SSISR.IIRQ = 1.

FRM[1:0] Bits

These bits set system word number in one frame of the communication formats.

Rewrite these bits when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 27.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

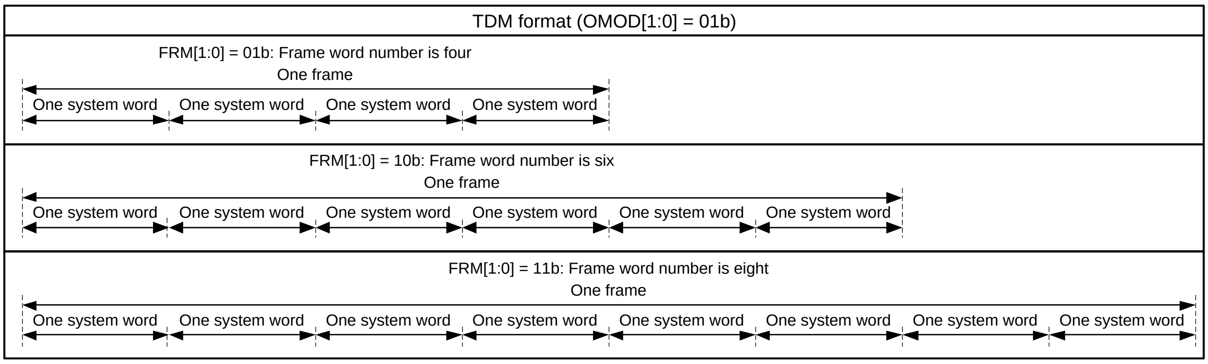


Figure 27.4 Frame word number

DWL[2:0] Bits

These bits set the number of bits in one data word. The setting whose number of data of system word lengths is shorter than that of the data word length is a prohibition. Refer to **Table 27.13** in **Section 27.4.2, Communication Formats** for details.

SWL[2:0] Bits

These bits set the number of bits in one system word. Padding bits are sent and received in relation with one data word set with DWL[2:0]. See **Table 27.13** in **Section 27.4.2, Communication Formats** for details.

Rewrite these bits when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 27.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

MST Bit

This bit sets master-/slave-mode communication.

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 27.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

BCKP Bit

This bit sets the bit clock polarity (**Table 27.5**).

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 27.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

Table 27.5 Bit Clock Polarity

Communication	Master/Slave	Timing	BCKP = 0	BCKP = 1
Reception	Slave	At SSILRCK/SSIFS sampling	SSIBCK rising edge	SSIBCK falling edge
	Master/slave	At SSIFRxD sampling	SSIBCK rising edge	SSIBCK falling edge
Transmission	Master	At change of SSILRCK/SSIFS output	SSIBCK falling edge	SSIBCK rising edge
	Master/slave	At change of SSIFTxD output	SSIBCK falling edge	SSIBCK rising edge

LRCKP Bit

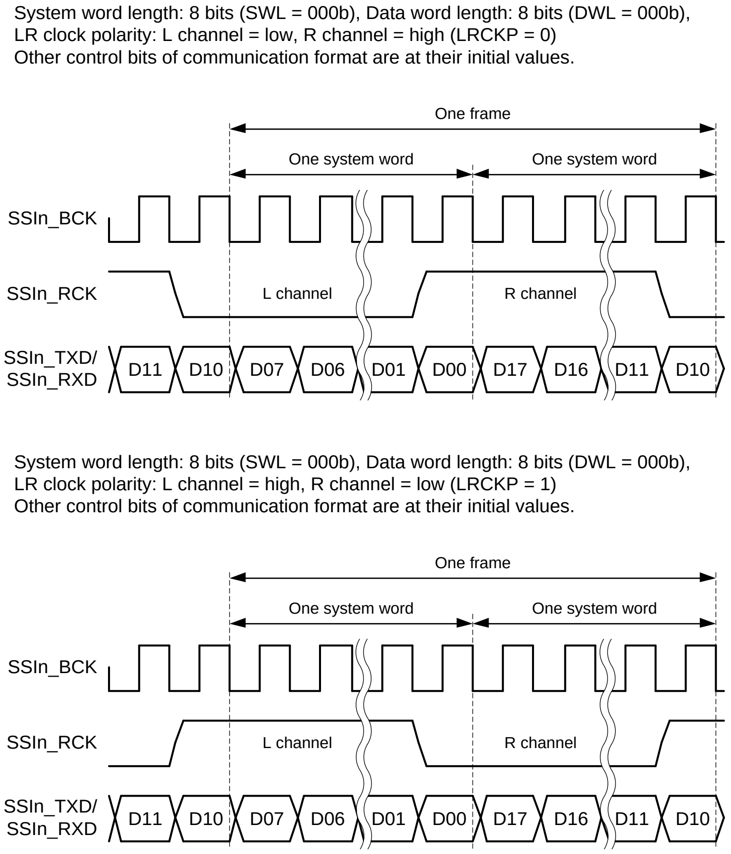
This bit sets the initial value and polarity of SSILRCK/SSIFS. Set this bit according to the communication format to be used in SSIF-2 (**Table 27.6**). Only the start trigger is used at slave communication (MST=0).

Rewrite this bit when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 27.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

Table 27.6 Initial Output Value and Polarity of SSILRCK/SSIFS Pin

Communication Format	Expected Initial State	Setting Value of LRCKP
I^{2}S	H	0
Monaural	L	1
TDM	L	1

Note: Do the setting that can be communicated respectively to I^{2}S, monaural, and TDM at the compatible format.



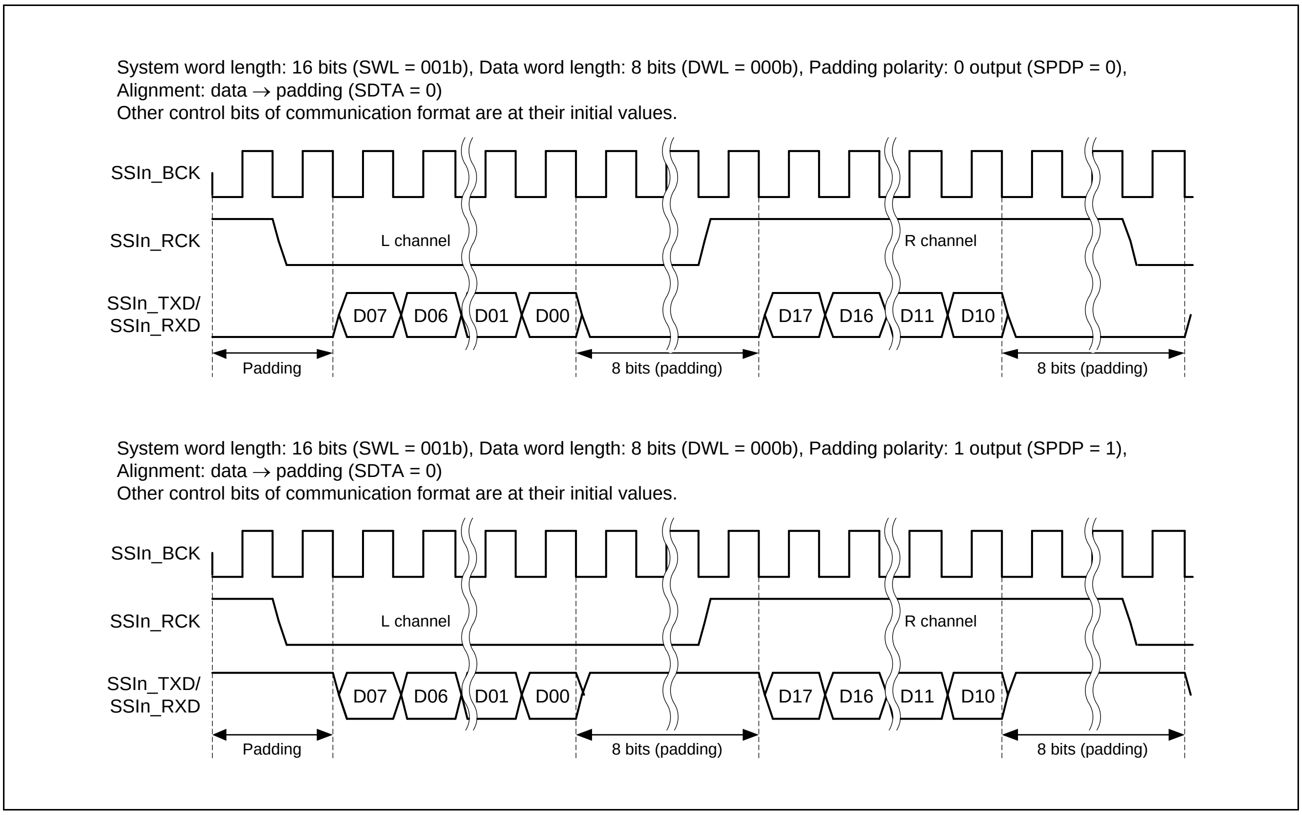
System word length: 8 bits (SWL = 000b), Data word length: 8 bits (DWL = 000b),
LR clock polarity: L channel = low, R channel = high (LRCKP = 0)
Other control bits of communication format are at their initial values.

System word length: 8 bits (SWL = 000b), Data word length: 8 bits (DWL = 000b),
LR clock polarity: L channel = high, R channel = low (LRCKP = 1)
Other control bits of communication format are at their initial values.

Figure 27.5 LRCK/FS Polarity Setting

SPDP Bit

This bit sets polarity of padding bits.


System word length: 16 bits (SWL = 001b), Data word length: 8 bits (DWL = 000b), Padding polarity: 0 output (SPDP = 0),
Alignment: data → padding (SDTA = 0)
Other control bits of communication format are at their initial values.

System word length: 16 bits (SWL = 001b), Data word length: 8 bits (DWL = 000b), Padding polarity: 1 output (SPDP = 1),
Alignment: data → padding (SDTA = 0)
Other control bits of communication format are at their initial values.

Figure 27.6 Padding Bit Polarity

SDTA Bit

This bit sets alignment of serial data (data bits and padding bits). For communication without padding bits, this bit is invalid.

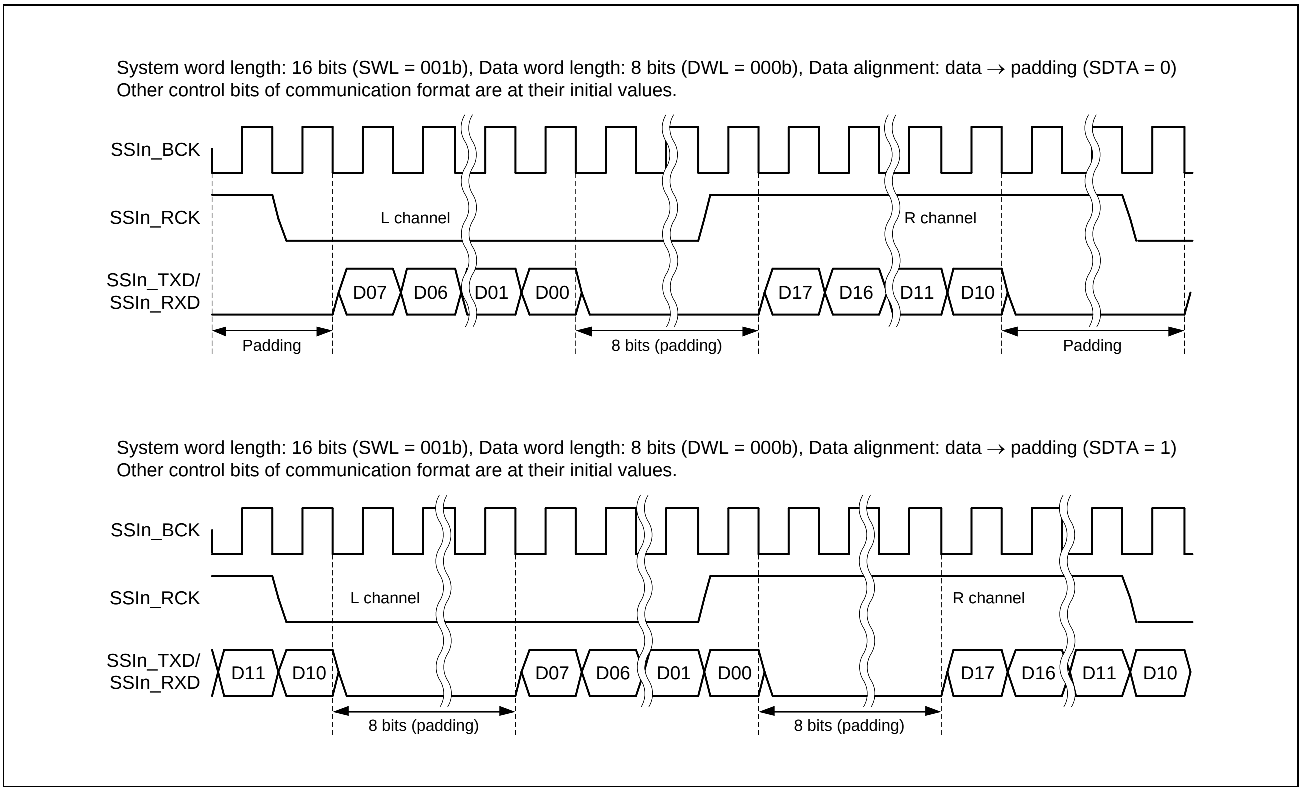
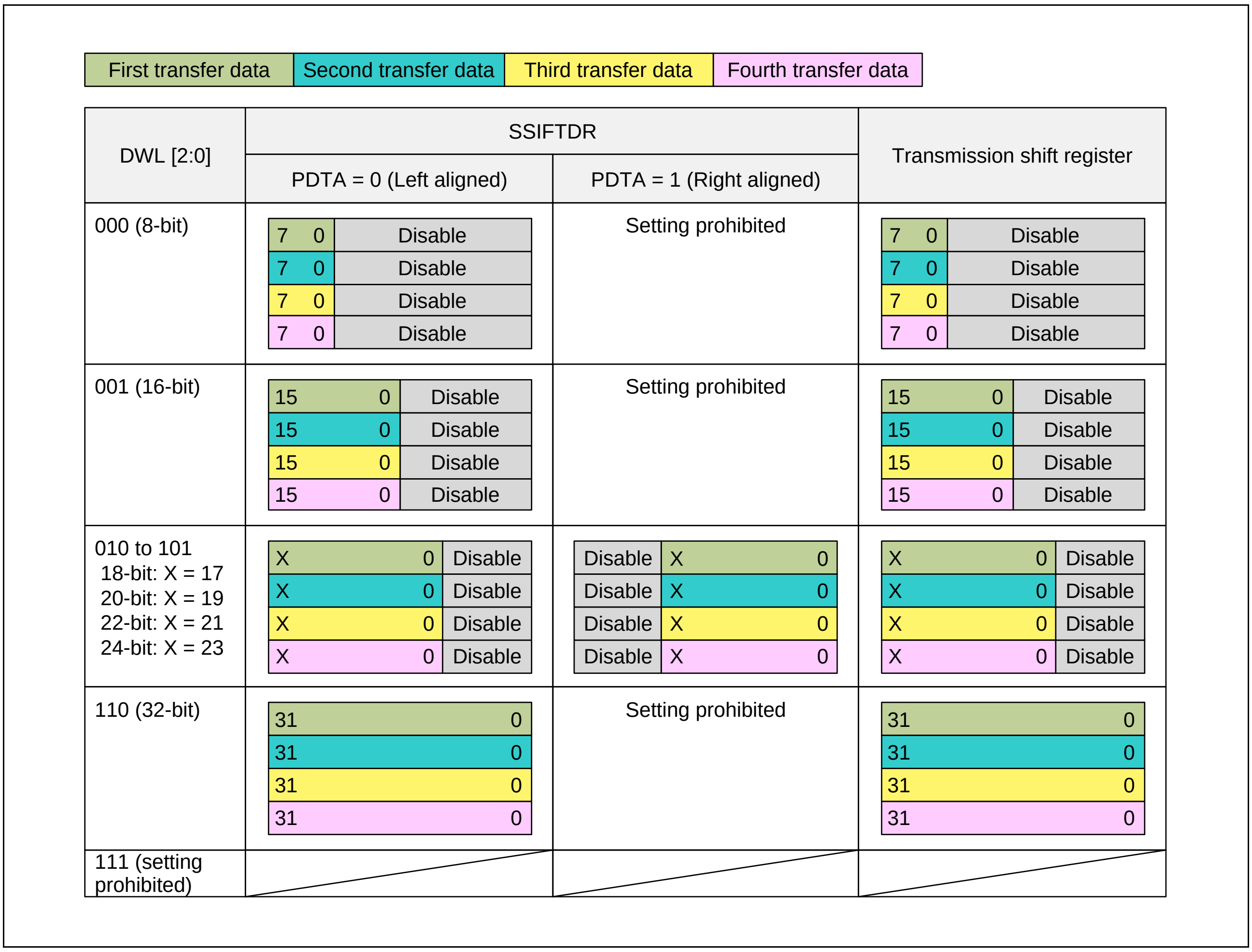


Figure 27.7 Alignment Setting of Serial Data with Padding Bits

PDTA Bit

This bit sets alignment of placement data. With the setting of data word length as 32 bits (DWL[2:0] = 110b), this bit is invalid.

At transmission:



First transfer data | Second transfer data | Third transfer data | Fourth transfer data

DWL [2:0]	SSIFTDR		Transmission shift register
	PDTA = 0 (Left aligned)	PDTA = 1 (Right aligned)	
000 (8-bit)	7 0 Disable 7 0 Disable 7 0 Disable 7 0 Disable	Setting prohibited	7 0 Disable 7 0 Disable 7 0 Disable 7 0 Disable
001 (16-bit)	15 0 Disable 15 0 Disable 15 0 Disable 15 0 Disable	Setting prohibited	15 0 Disable 15 0 Disable 15 0 Disable 15 0 Disable
010 to 101 18-bit: X = 17 20-bit: X = 19 22-bit: X = 21 24-bit: X = 23	X 0 Disable X 0 Disable X 0 Disable X 0 Disable	Disable X 0 Disable X 0 Disable X 0 Disable X 0	X 0 Disable X 0 Disable X 0 Disable X 0 Disable
110 (32-bit)	31 0 31 0 31 0 31 0	Setting prohibited	31 0 31 0 31 0 31 0
111 (setting prohibited)			

Figure 27.8 Alignment Setting of Placement Data in Transmission

At reception:

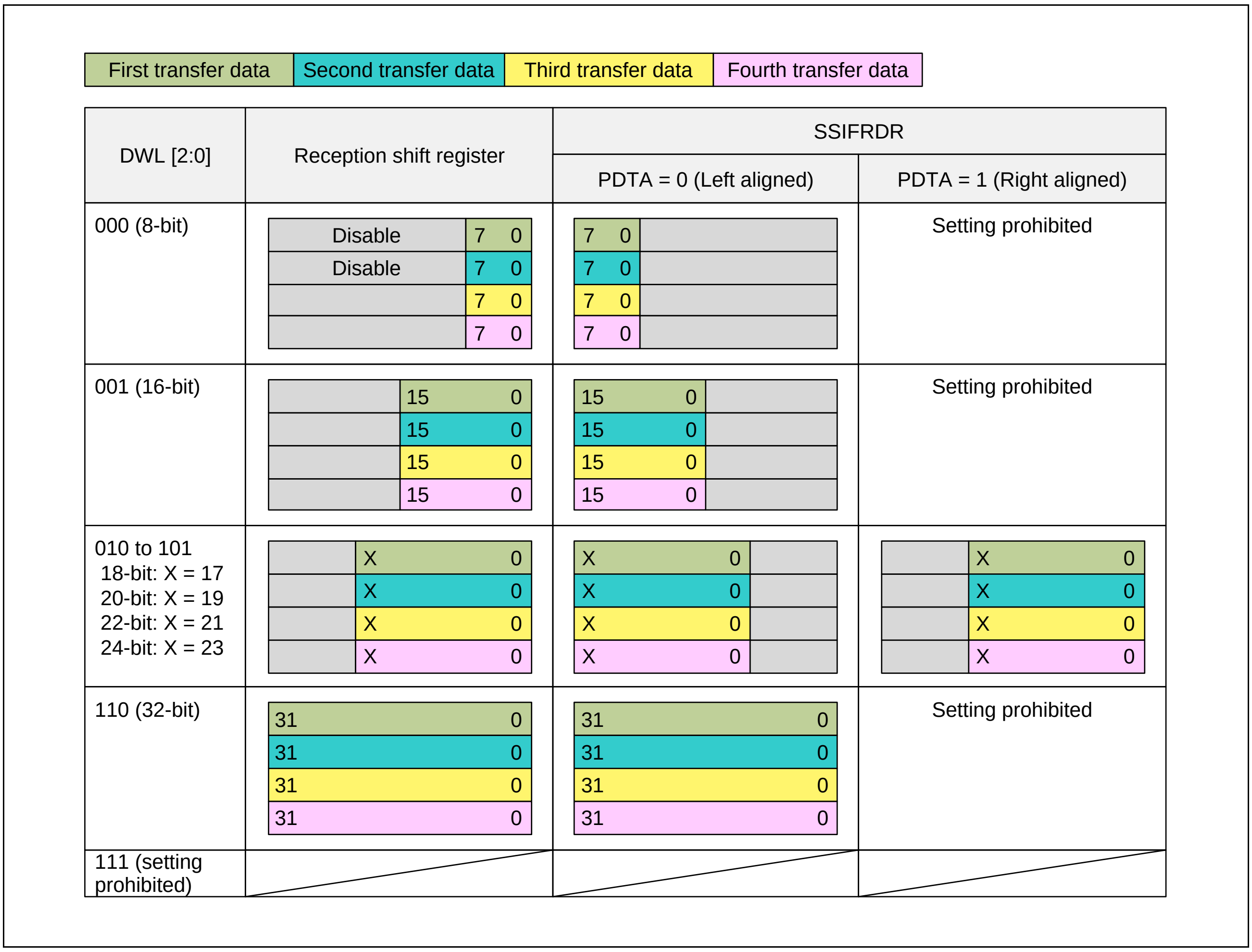


Figure 27.9 Alignment Setting of Placement Data in Reception

DEL Bit

This bit sets whether or not there will be a delay between SSILRCK/SSIFS and SSITxD/SSIRxD.

With the I²S format and TDM format, set 0 to DEL (**Figure 27.10**). With the monaural format, this bit changes the high period width. Refer to **Section 27.4.2.2, Monaural Format** for details. Do the setting that can be communicated at the compatible communication format.

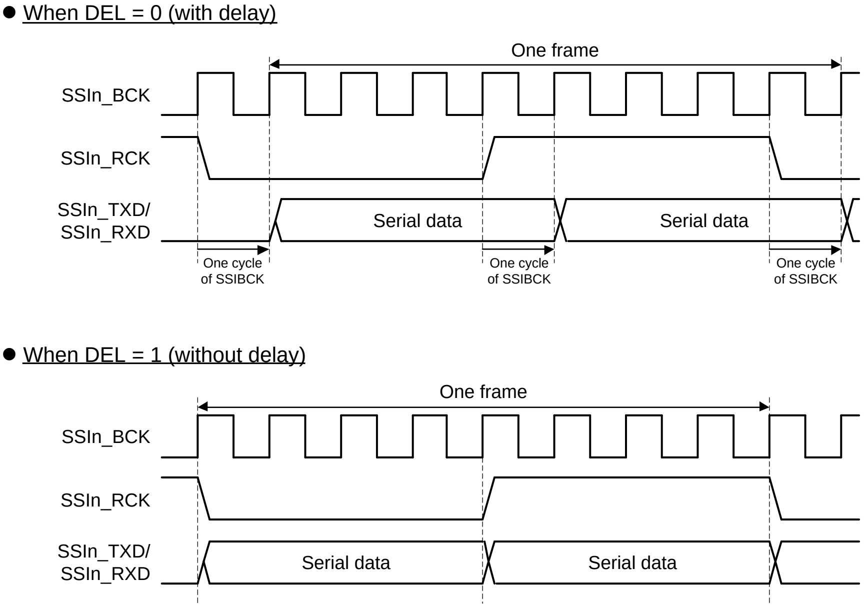


Figure 27.10 Setting of Delay in Serial Data

CKDV[3:0] Bits

These bits set the division ratio of the bit clock based on AUDIO_MCK in master-mode communication (MST = 1) (**Figure 27.11**). In slave-mode communication (MST = 0), setting of these bits are invalid.

To write this bit, do so when AUDIO_MCK is not supplied. Refer to the AUCKE bit details explanation in **Section 27.4.1.3, FIFO Control Register (SSIFCR)** for detailed timing.

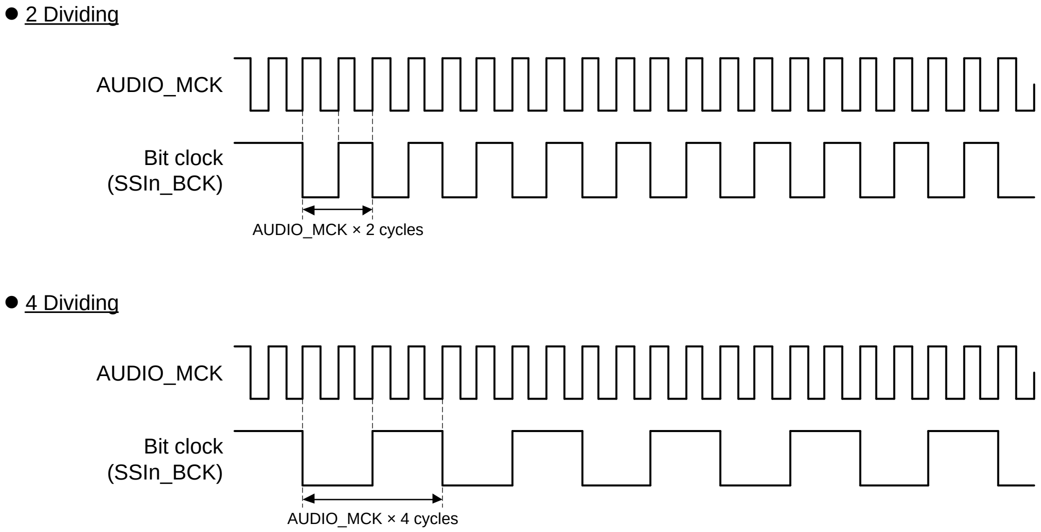


Figure 27.11 Sampling Frequencies in Master-mode Communication

MUEN Bit

This bit sets/clears the mute function for the data output from the terminal SSITxD. When you set this bit to 1, the output value of SSITxD becomes 0 from the next frame boundary. When you set this bit to 0, the output value of SSITxD can be made the data of the FIFO transmission data register from the following frame boundary.

Note that this bit controls data only. Status flags and interrupt signals are normally generated. Set this bit after setting completing the communication format.

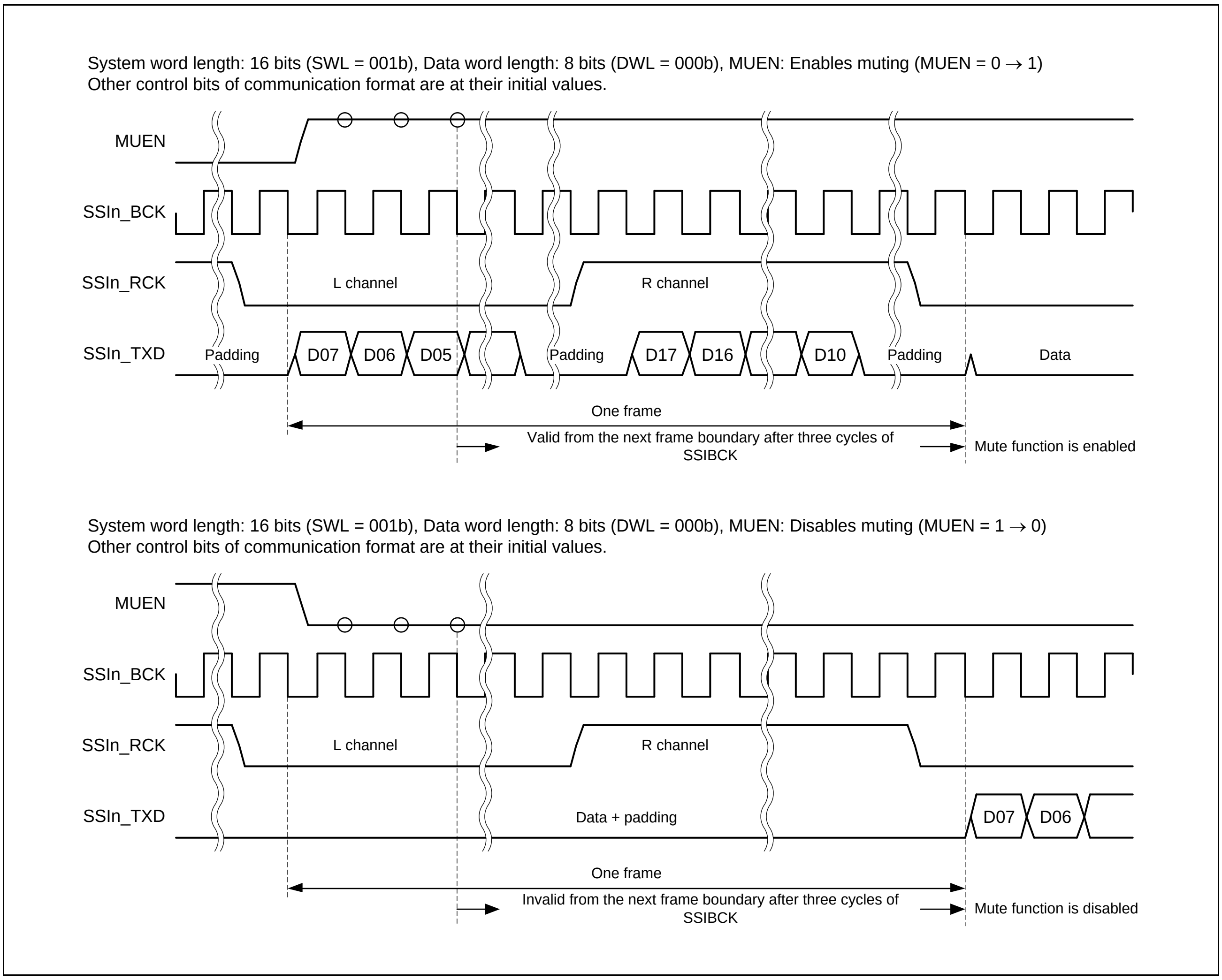


Figure 27.12 Transmit Data with the Mute Function Set

TEN and REN Bits

These bits enable/disable transmission and reception. Writing 1 to these bits, transmission starts in synchronization with start trigger. Refer to **Section 27.5.2.2, Transmission**, **Section 27.5.2.3, Reception**, and **Section 27.5.2.4, Transmission and Reception** for details. Writing 0 to these bits, transmission stops in synchronization with next frame boundary. Set 1 at the same time when you use SSIF-2 as transmission and reception. To stop communication, make sure to stop transmission and reception (TEN = 0 and REN = 0).

To stop communication without waiting for the frame boundary, follow the software reset procedure.

27.4.1.2 Status Register (SSISR)

This is a readable/writable 32-bit register. It is configured with status flags that indicate SSIF-2 operational state.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W0	R/W0	R/W0	R/W0	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved. Write 0. The read value is 0.
29	TUIRQ	0	RW0	Transmit underflow error status flag 0: No transmit underflow error is generated. 1: A transmit underflow error is generated.
28	TOIRQ	0	RW0	Transmit overflow error status flag 0: No transmit overflow error is generated. 1: A transmit overflow error is generated.
27	RUIRQ	0	RW0	Receive underflow error status flag 0: No receive underflow error is generated. 1: A receive underflow error is generated.
26	ROIRQ	0	RW0	Receive overflow error status flag 0: No receive overflow error is generated. 1: A receive overflow error is generated.
25	IIRQ	1	R	Idle mode status flag 0: In the communication state 1: In the idle state
24 to 0	—	H’0000000	R	Reserved. Write 0. The read value is 0.

TUIRQ Bit

This is a status flag that indicates a transmit underflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that writing serial data necessary for one frame to SSIFTDR was slower than one frame of the transmission operation. Even if this flag is cleared after this flag becomes ‘1’, the output value of SSIFTxD is 0. Follow the communication stop procedure (**Figure 27.55**) and the error procedure (**Figure 27.56**) to output the data written in transmission FIFO data register (SSIFTDR) to the terminal SSIFTxD. See **Section 27.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by transmission FIFO data register reset (SSIFCR.TFRST).

[Priority order for setting and clearing]

Setting is prioritized.*[1]

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*[2]

(2) When you do the communication permission (SSICR.TEN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (**Figure 27.13**).

(2) Write 1 in SSICR.TEN, complete, and after one cycle of P0ϕ (**Figure 27.13**).

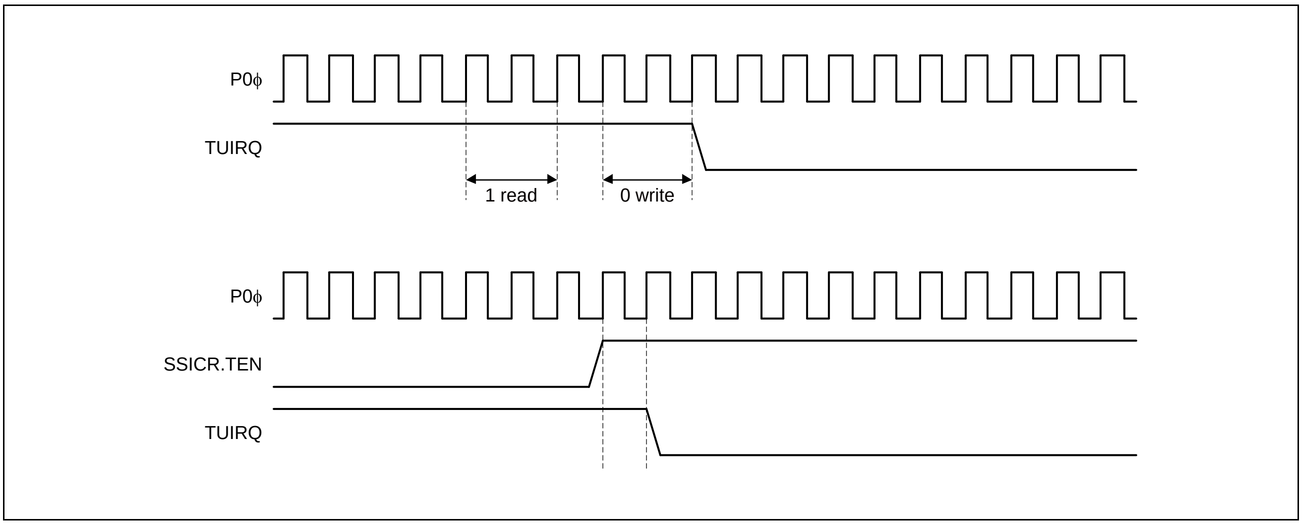


Figure 27.13 TUIRQ Clearing Timing

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

NOTE

After the communication is permitted (SSICR.TEN is written and ‘1’ is written from ‘0’), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

[Setting condition]

When data to be transmitted in the next frame is not fully written to SSIFTDR at the frame boundary of continuance communication.

[Setting timing]

After three cycles of P0ϕ from frame boundary (**Figure 27.14**).

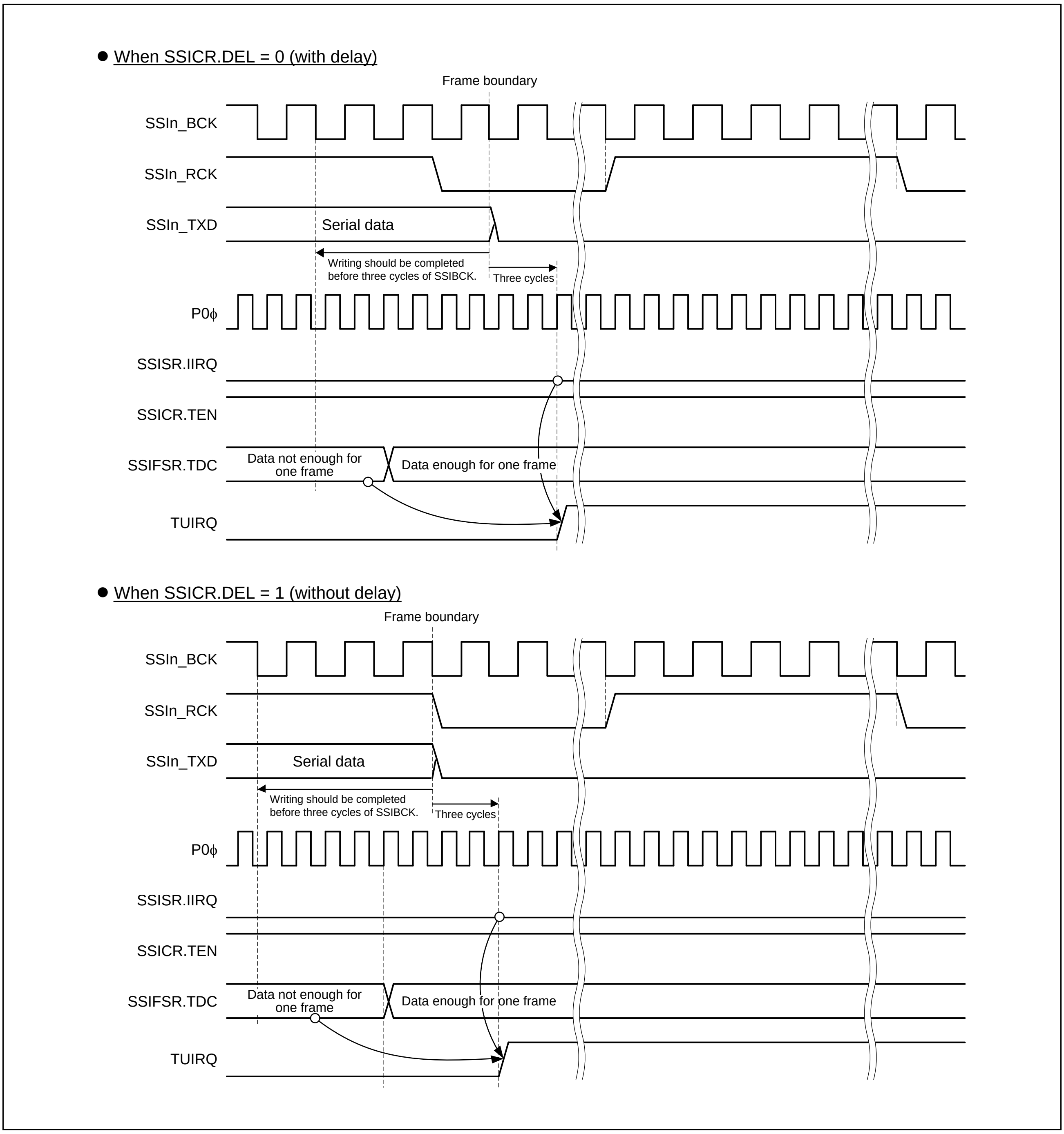


Figure 27.14 TUIRQ Setting Timing (When you continue communicating)

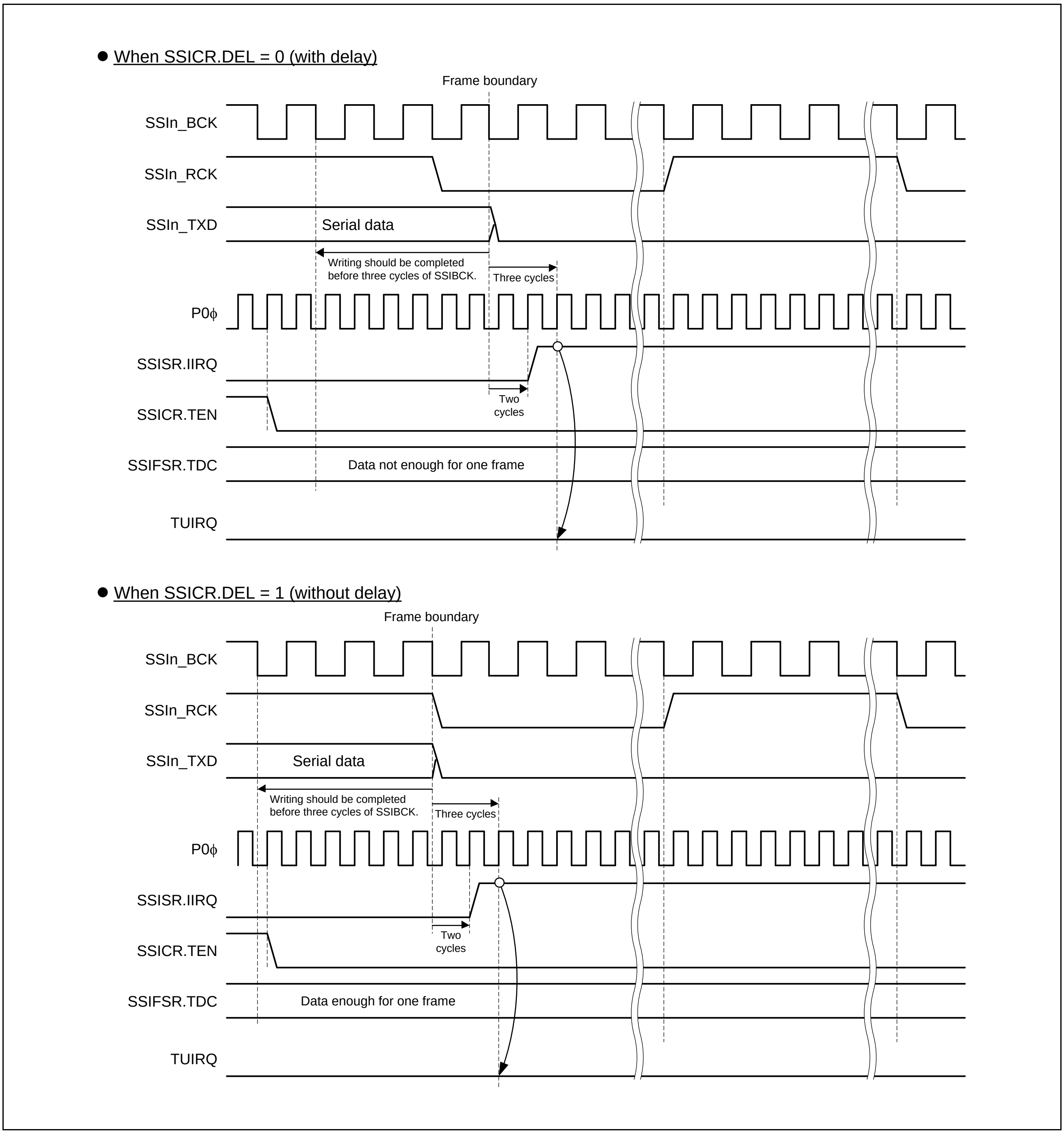


Figure 27.15 TUIRQ Setting Timing (When you stop communicating)

TOIRQ Bit

This is a status flag that indicates a transmit overflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that serial data is written to SSIFTDR when it is full during transmission (SSICR.TEN=1). Data is not written to SSIFTDR where a transmit overflow error is generated. See **Section 27.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by transmission FIFO data register reset (SSIFCR.TFRST).

[Priority order for setting and clearing]

Setting is prioritized.*[1]

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*[2]

(2) When you do the communication permission (SSICR.TEN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as **Figure 27.13**).

(2) Write 1 in SSICR.TEN, complete, and after one cycle of P0ϕ (same as **Figure 27.13**).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

NOTE

After the communication is permitted (SSICR.TEN is written and ‘1’ is written from ‘0’), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

[Setting condition]

During transmission operation (SSICR.TEN = 1), data is written to SSIFTDR while it is full.

[Setting timing]

At completion of writing to SSIFTDR (**Figure 27.16**)

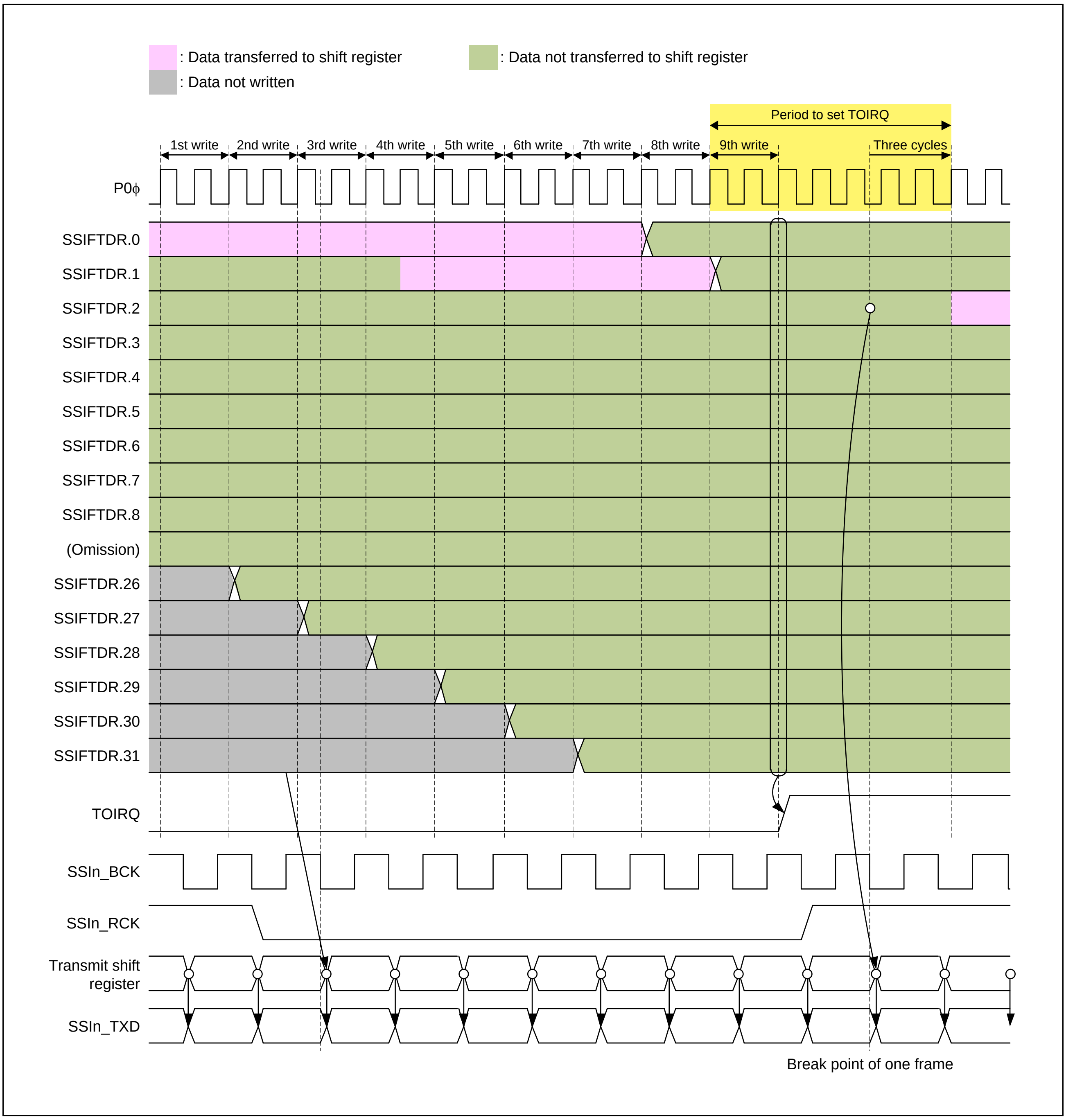


Figure 27.16 TOIRQ Setting Timing

RUIRQ Bit

This is a status flag that indicates a receive underflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that SSIFRDR is read while it is empty. Data read from SSIFRDR where a receive underflow error is generated is invalid. See **Section 27.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by reception FIFO data register reset (SSIFCR.RFRST). Even if SSIFRDR is read to (SSIFCR.RFRST=1) while resetting reception FIFO data register, it doesn’t set 1.

[Priority order for setting and clearing]

Setting is prioritized.*[1]

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*[2]

(2) When you do the communication permission (SSICR.REN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as **Figure 27.13**).

(2) Write 1 in SSICR.REN, complete, and after one cycle of P0ϕ (same as **Figure 27.13**).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

NOTE

After the communication is permitted (SSICR.REN is written and ‘1’ is written from ‘0’), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

[Setting condition]

Reading from SSIFRDR while it is empty.

[Setting timing]

At completion of reading from SSIFRDR (**Figure 27.17**).

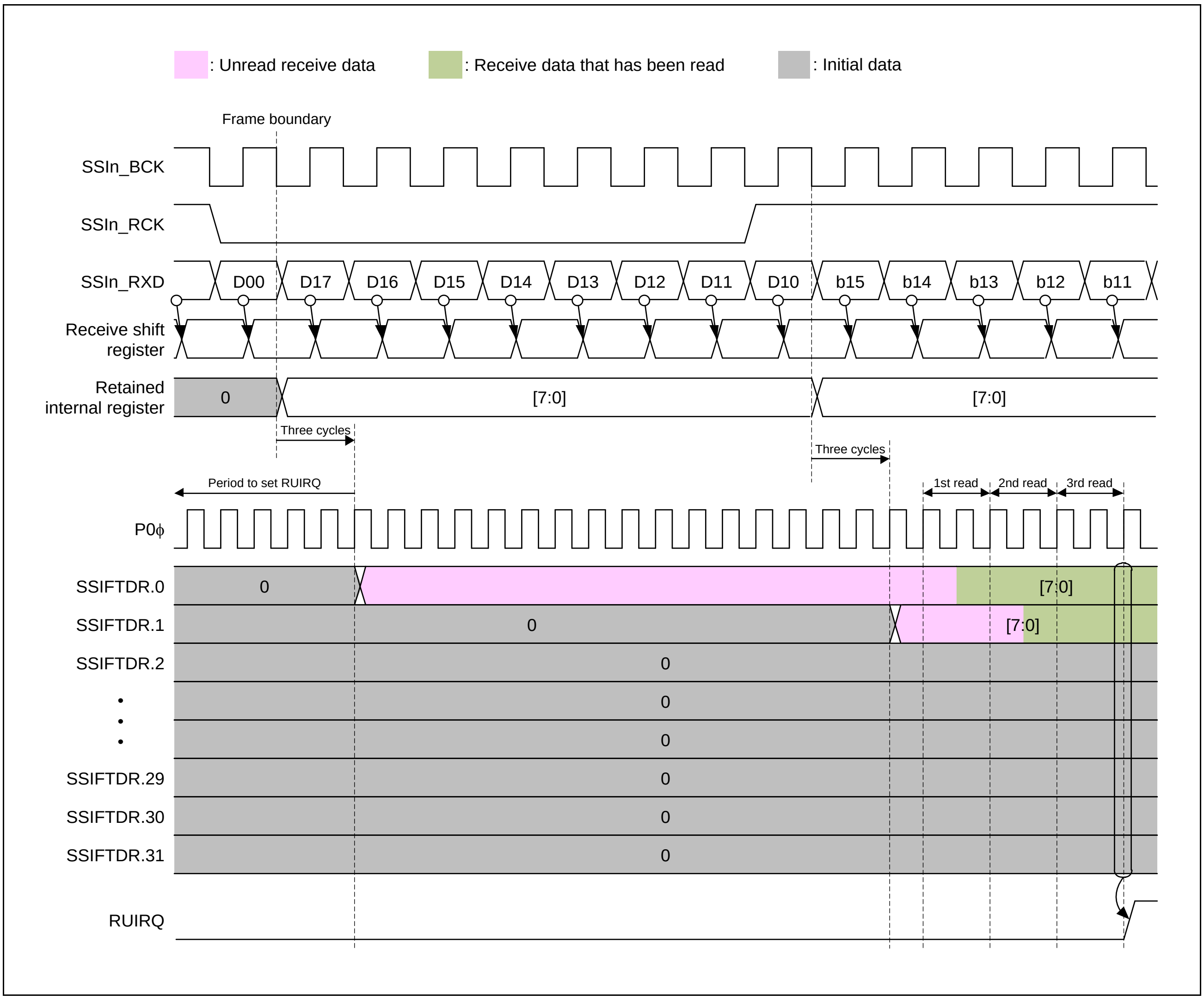


Figure 27.17 RUIRQ Setting Timing

ROIRQ Bit

This is a status flag that indicates a receive overflow error. This flag is set by automatic determination but it must be cleared by register access. This flag indicates that received data is supplied at a higher rate than requested. Data is not transferred from the receive shift register to SSIFRDR where a receive overflow error is generated. See **Section 27.5.2.6, Error Handling** for the error recovery procedure. This flag is not cleared by reception FIFO data register reset (SSIFCR.RFRST).

[Priority order for setting and clearing]

Setting is prioritized.

[Clearing condition]

Writing 0 to this flag after reading this flag as 1.

[Clearing timing]

At completion of writing 0 after reading 1 from an SFR (same as **Figure 27.13**).

[Priority order for setting and clearing]

Setting is prioritized.*[1]

[Clearing condition]

When either of the following is approved.

(1) Writing 0 to this flag after reading this flag as 1.*[2]

(2) When you do the communication permission (SSICR.REN is changed from 0 to 1)

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this flag after reading this flag as 1 (same as **Figure 27.13**).

(2) Write 1 in SSICR.REN, complete, and after one cycle of P0ϕ (same as **Figure 27.13**).

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1). The software reset has priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following three conditions being met:
- Software reset (SSIFCR.SSIRST=1).
- Completion of writing 0 to this flag after having read it as 1.
- One cycle of P0ϕ elapsing after the completion of writing 1 to SSICR.TEN.

NOTE

After the communication is permitted (SSICR.REN is written and ‘1’ is written from ‘0’), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

[Setting condition]

At completion of receiving new data while SSIFRDR is full.

[Setting timing]

Three cycles of P0ϕ after reception is completed (**Figure 27.18**).

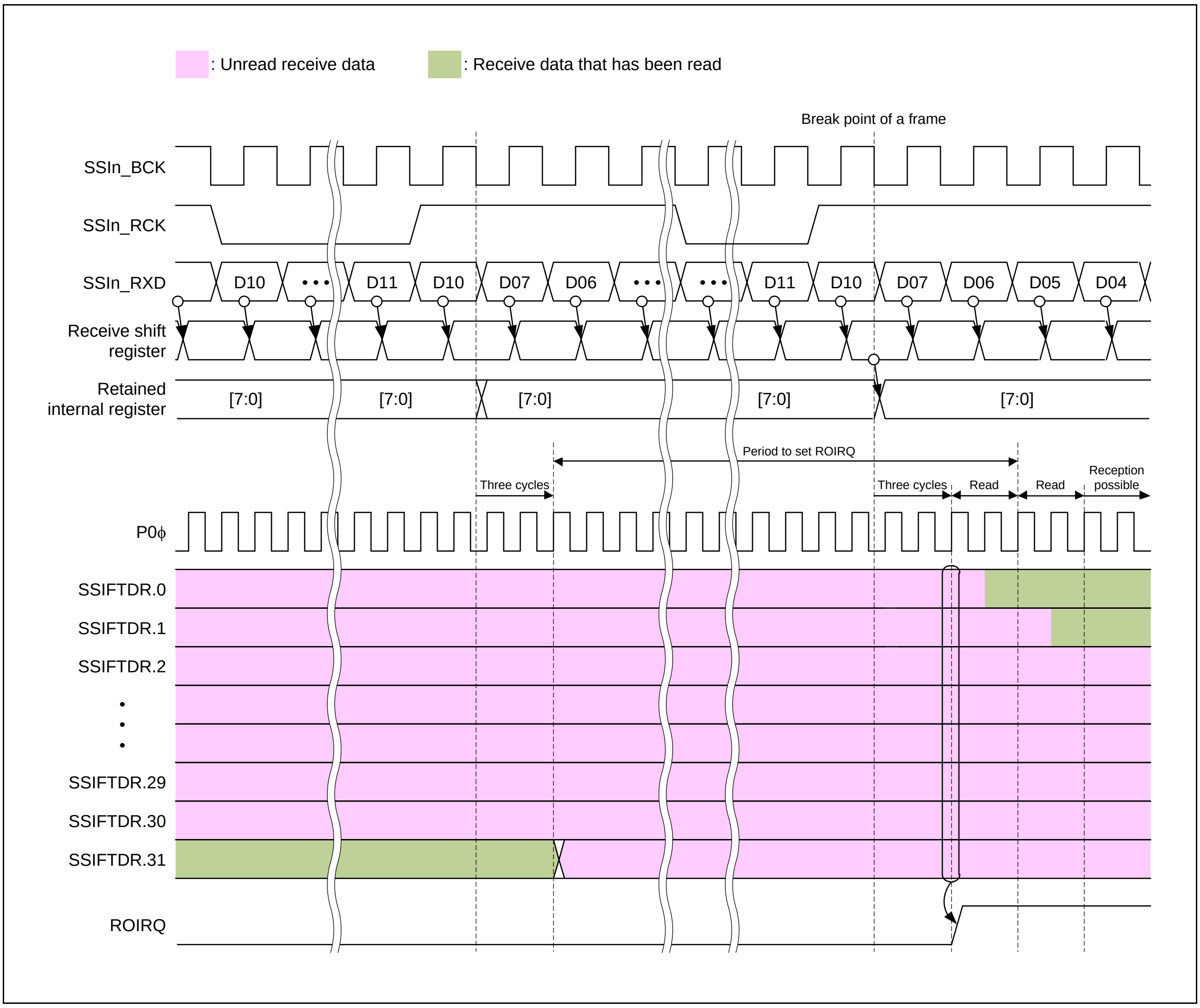


Figure 27.18 ROIRQ Setting Timing

IIRQ Bit

This is a status flag that indicates the idle state. It indicates whether SSIF-2 is in the idle state or communication state (**Figure 27.19** and **Figure 27.20**).

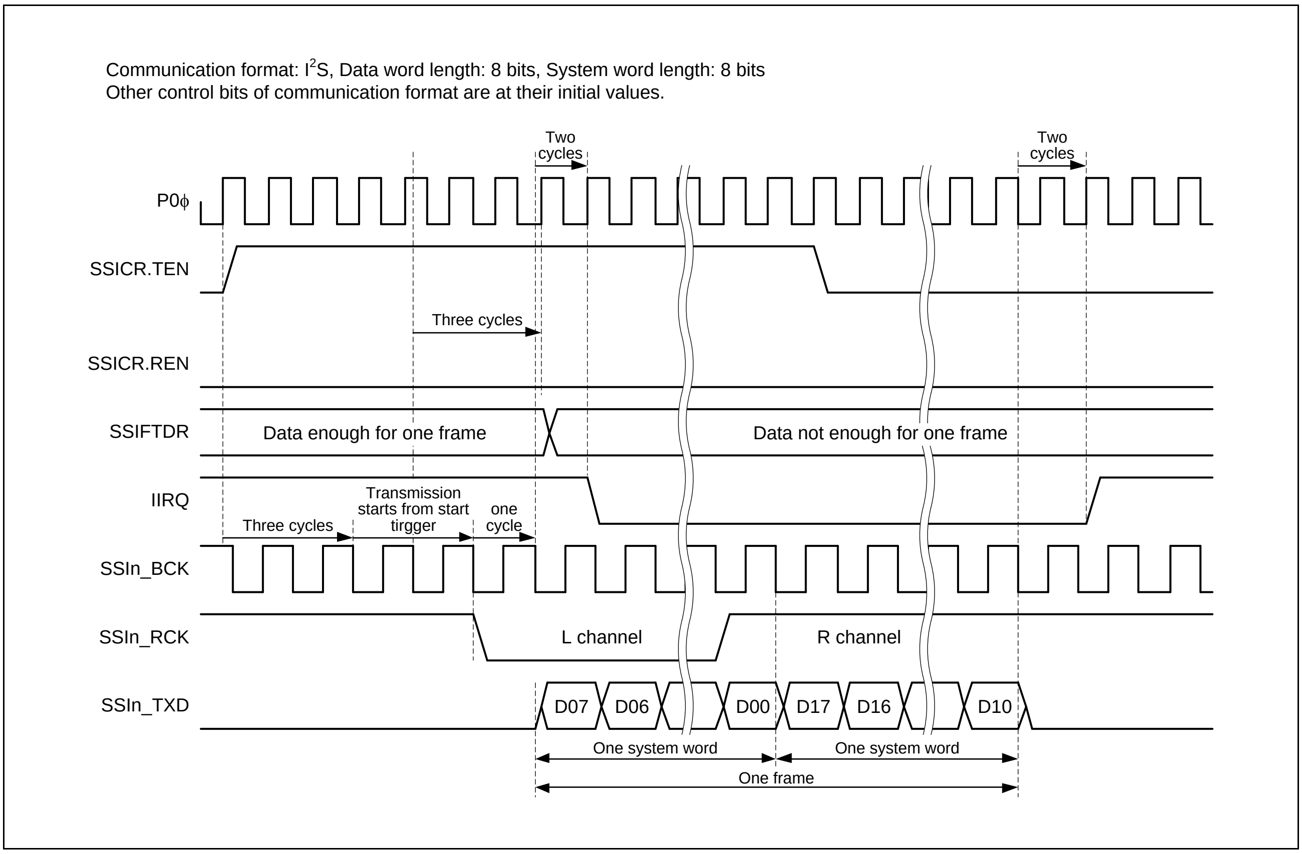


Figure 27.19 IIRQ Setting Timing (Transmission)

Transmitter (dedicated to transmission)

[Clearing condition]

While transmission is enabled (SSICR.TEN = 1 and SSICR.REN = 0), one frame of data to be transmitted has been written to SSIFTDR, and a start trigger is generated in SSILRCK/SSIFS.

[Clearing timing]

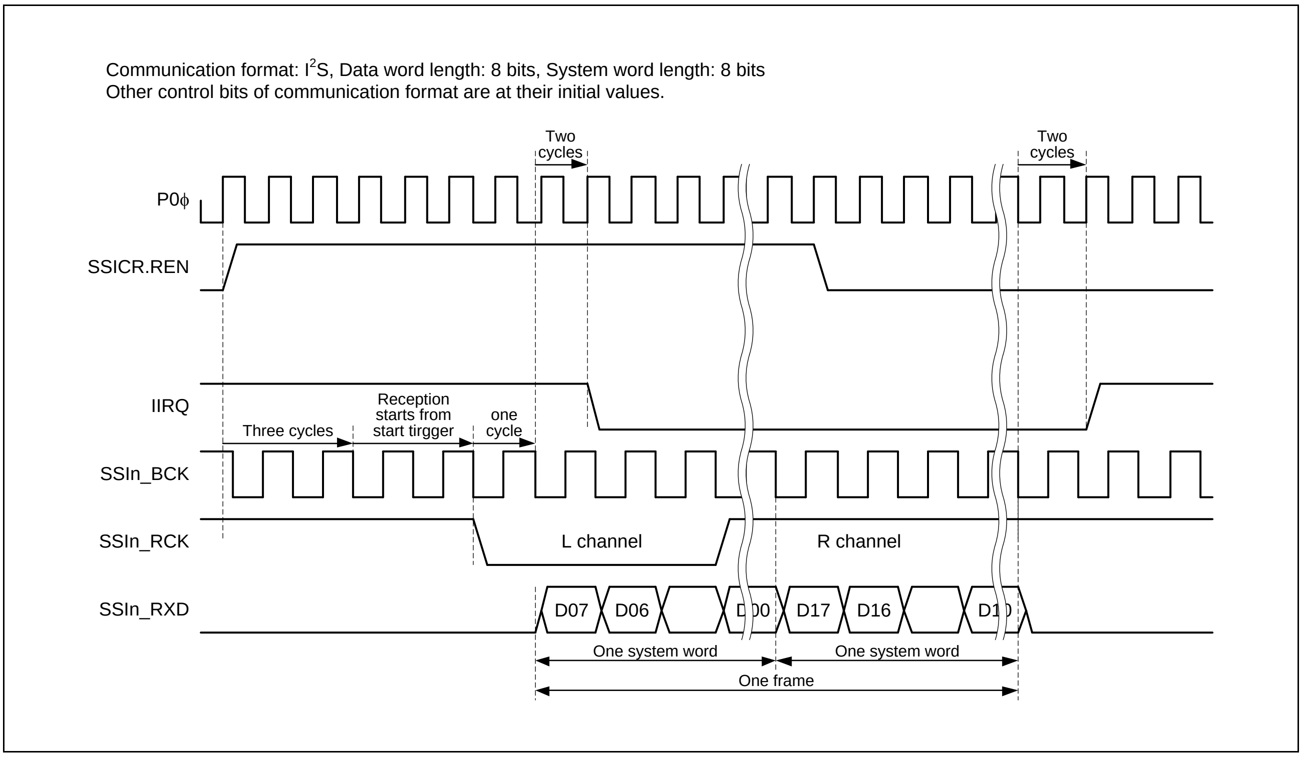
Two cycles of P0ϕ after one cycle of SSIBCK where a start trigger is generated.

[Setting condition]

One frame of data has been transmitted while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).

[Setting timing]

Two cycles of P0ϕ after transmission is complete according to the setting condition (frame boundary).



Communication format: I^2S, Data word length: 8 bits, System word length: 8 bits
Other control bits of communication format are at their initial values.

Figure 27.20 IIRQ Setting Timing (Reception)

Receiver (dedicated to reception)

[Clearing condition]

While reception is enabled (SSICR.TEN = 0 and SSICR.REN = 1), and a start trigger is generated in SSILRCK/SSIFS.

[Clearing timing]

Two cycles of P0ϕ after one cycle of SSIBCK when a start trigger of the clearing condition is generated.

[Setting condition]

One frame of data has been received while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).

[Setting timing]

Two cycles of P0ϕ after reception of the setting condition is completed (frame boundary).

Transceiver (transmission and reception)

[Clearing condition]

While transmission and reception are enabled (SSICR.TEN = 1 and SSICR.REN = 1), one frame of data to be transmitted has been written to SSIFTDR, and a start trigger is generated in SSILRCK/SSIFS.

[Clearing timing]

Two cycles of P0ϕ after one cycle of SSIBCK when a start trigger of the clearing condition is generated.

[Setting condition]

One frame of data has been transmitted while transmission and reception are disabled (SSICR.TEN = 0 and SSICR.REN = 0).

[Setting timing]

Two cycles of P0ϕ after transmission is completed according to the setting condition (frame boundary).

27.4.1.3 FIFO Control Register (SSIFCR)

This is a readable/writable 32-bit register. It sets a software reset, byte swap, noise canceller, and enable/disable of interrupt requests.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BSW	BCKNCE	LRCKNCE	RXDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	AUCKE	0	RW	AUDIO_MCK Enable in Master-mode Communication*1 0: Disables supply of AUDIO_MCK 1: Enables supply of AUDIO_MCK
30 to 17	—	H’0000	R	Reserved. Write 0. The read value is 0.
16	SSIRST	0	RW	Software Reset 0: Clears a software reset condition 1: Sets a software reset condition
15 to 12	—	H’00	R	Reserved. Write 0. The read value is 0.
11	BSW	0	RW	Byte Swap Enable*1 0: Disables byte swap 1: Enables byte swap
10	BCKNCE	0	RW	Noise Canceller Enable in Slave-mode Communication (SSIBCK)*1,*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
9	LRCKNCE	0	RW	Noise Canceller Enable in Slave-mode Communication (SSILRCK/ SSIFS)*1,*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
8	RXDNCE	0	RW	Received Data Input Noise Canceller Enable in Slave-mode Communication (SSIRxD)*1,*2 0: Disables (bypasses) the noise canceller 1: Enables the noise canceller
7 to 4	—	H’00	R	Reserved. Write 0. The read value is 0.
3	TIE	0	RW	Transmit Data Empty Interrupt Output Enable 0: Disables transmit data empty interrupts 1: Enables transmit data empty interrupts
2	RIE	0	RW	Receive Data Full Interrupt Output Enable 0: Disables receive data full interrupts 1: Enables receive data full interrupts
1	TFRST	0	RW	Transmit FIFO data register reset*1 0: Clears a transmit data FIFO reset condition 1: Sets a transmit data FIFO reset condition
0	RFRST	0	RW	Receive FIFO data register reset*1 0: Clears a receive data FIFO reset condition 1: Sets a receive data FIFO reset condition

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Note 2. This bit can only be set to 1 in slave-mode communication (SSICR.MST = ‘0’) except in the case of transmission (SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Be sure to clear this bit to 0 in master-mode communication (SSICR.MST = ‘1’) or during transmission in slave mode (SSICR.MST = ‘0’, SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

AUCKE Bit

This bit enables/disables supply to AUDIO_MCK while in master-mode communication (MST = 1). Select AUDIO_MCK (SSICR.CKS bit) before writing 1 to this bit.

Rewrite this bit after completing the setting (CKS, MST, BCKP, and CKDV of the SSICR register) that relates to AUDIO_MCK.

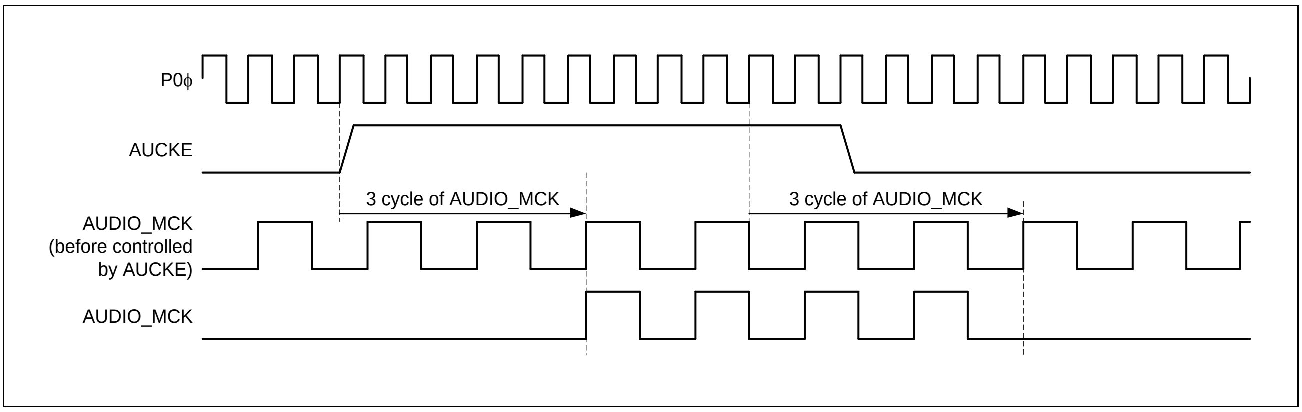


Figure 27.21 Stop/Resume of AUDIO_MCK

CAUTION

- In slave-mode communication (SSICR.MST = 0), SSIF-2 needs supply of SSIBCK. To stop BCK on the master side, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1). If BCK is stopped before SSIF-2 becomes idle, take the procedure to start communication (**Figure 27.52**) or wait for an idle state by taking the procedure to resume communication (**Figure 27.57**).
- In master-mode communication (SSICR.MST = 1), SSIF-2 operates with the audio clock (AUDIO_MCK). To stop SSIF-2 completely, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1) and then write 0 to SSIFCR.ADCKE. If 0 is written to SSIFCR.ADCKE before SSIF-2 becomes idle, take the procedure to start communication (**Figure 27.52**) or wait for the idle state by taking the procedure to resume communication (**Figure 27.57**).

Figure 27.22 and **Figure 27.23** show timing until outputting it to the terminal SSIBCK after this bit is set to ‘1’.

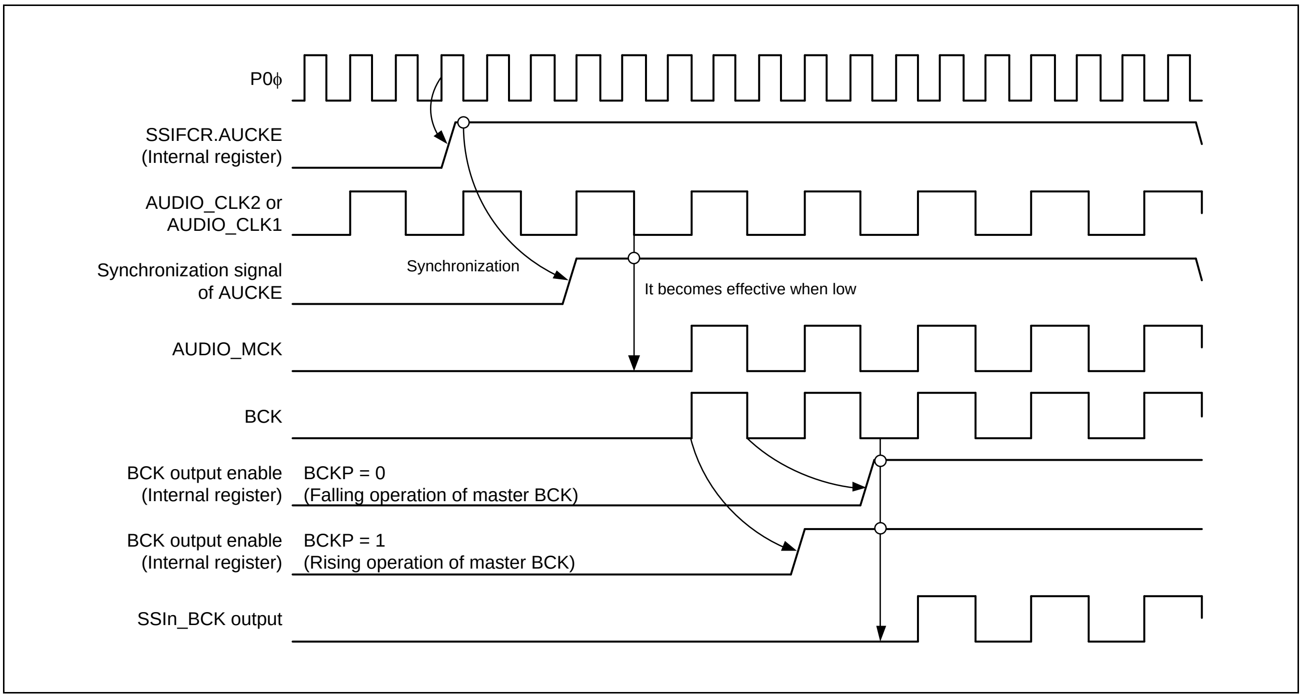


Figure 27.22 Timing chart when it begins to communicate master from system reset

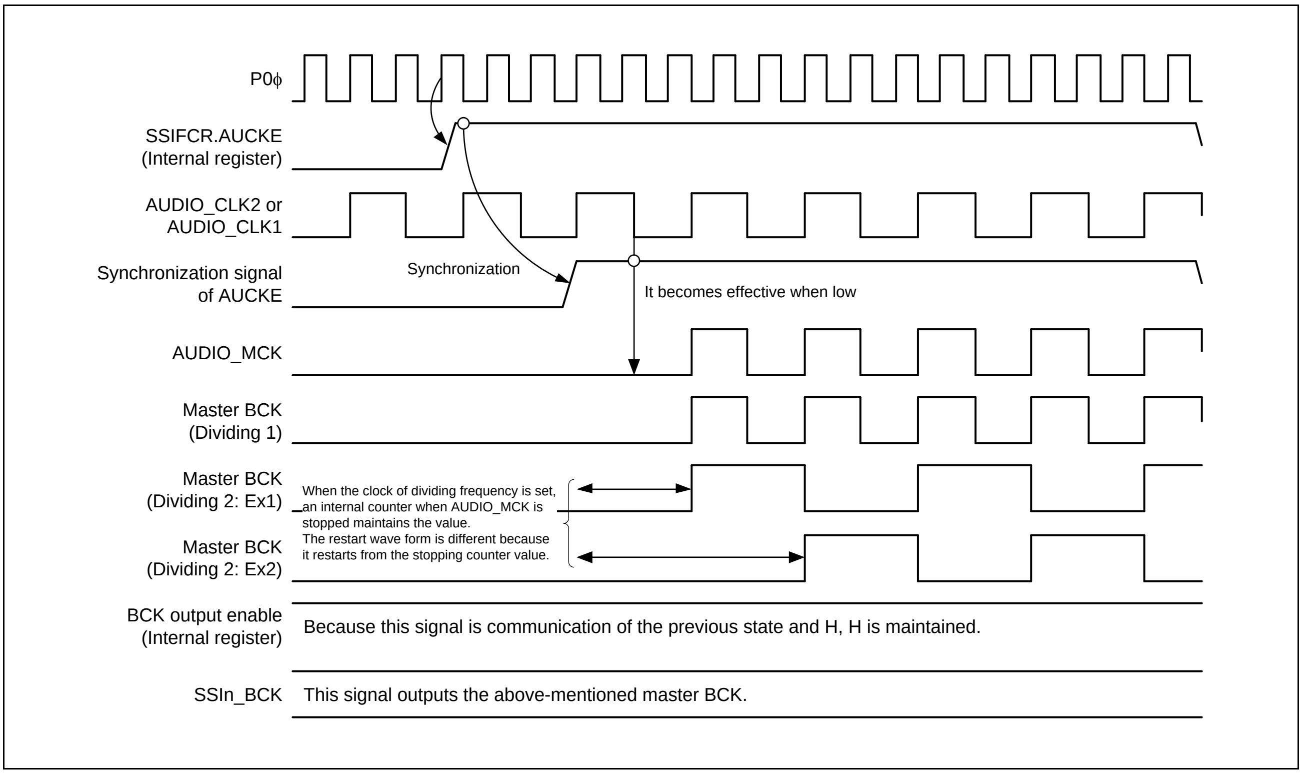


Figure 27.23 Timing chart when it begins to communicate master from communication stop

NOTE

When the supply of AUDIO_MCK is stopped, the terminal SSIBCK maintains the value. The terminal SSIBCK might stop in the state of “H”.

SSIRST Bit

This bit sets a software reset of SSIF-2. Writing 1 to this bit initializes the internal state of SSIF-2. After a reset by writing 1, write 0 to release the reset because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.

To stop communication of SSIF-2 immediately, after turning off the peripheral functions, write 1 to this bit. Initialization by a software reset is performed without any relation with the bit clock.

Table 27.7 Bits Initialized by Software Reset of the SSIFCR.SSIRST Bit

			+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol	Address (Base+)		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H’00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H’04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H’10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H’14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H’18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H’1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	H’20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H’24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

BSW Bit

This bit enables/disables byte swap of register access for the transmit FIFO data register (SSIFTDR) and the receive FIFO data register (SSIFRDR). This bit is valid only with 16-bit access or 32-bit access to SSIFTDR and SSIFRDR (**Figure 27.24**).

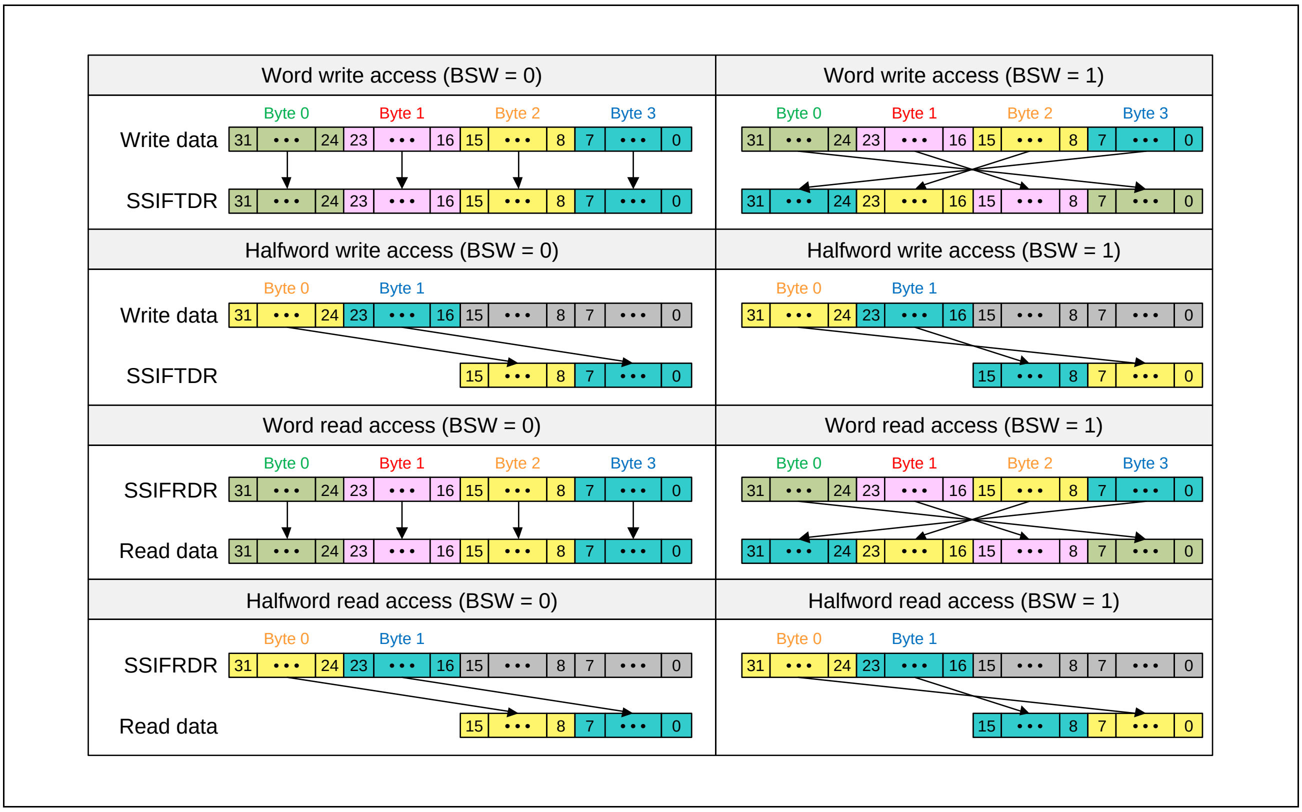


Figure 27.24 Operation Example of Byte Swap

BCKNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSIBCK pin in slave-mode communication (SSICR.MST = ‘0’) except in the case of transmission (SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. The setting of this bit in master-mode communication (SSICR.MST = ‘1’) is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = ‘0’, SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

LRCKNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSILRCK and SSIFS pins in slave-mode communication (SSICR.MST = ‘0’) except in the case of transmission (SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. The setting of this bit in master-mode communication (SSICR.MST = ‘1’) is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = ‘0’, SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

RXDNCE Bit

This bit enables or disables the noise cancellation function supporting the suppression of transients 15-ns in length (typ.) in input signals on the SSIRxD pin in slave-mode communication (SSICR.MST = ‘0’) except in the case of transmission (SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’ and while reception is enabled (SSICR.REN = ‘1’).

The setting of this bit in master-mode communication (SSICR.MST = ‘1’) or while reception is disabled (SSICR.REN = ‘0’) is disabled.

Be sure to clear this bit to 0 in master-mode communication or during transmission in slave mode (SSICR.MST = ‘0’, SSICR.TEN = ‘1’) while SSICR.DEL = ‘1’. Set the BCKNCE, LRCKNCE, and RXDNCE bits to the same value.

TIE Bit

This bit enables/disables output of transmit data empty interrupts. Transmission data empty interrupt is used as an interrupt factor written in the FIFO transmission data register. Set this bit to 1 after setting set condition (SSISCR.TDES) of transmission data empty interrupt. See **Figure 27.25** for the generation timing of a transmit data empty interrupt.

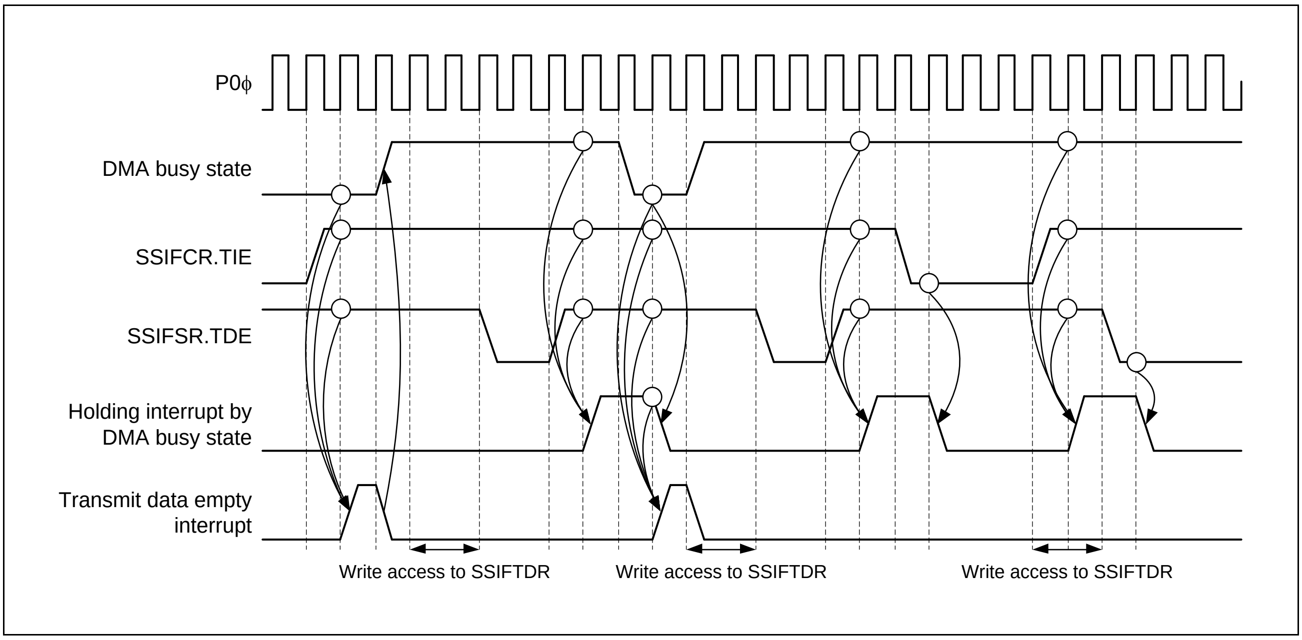


Figure 27.25 Generation timing of transmit data empty interrupt

RIE Bit

This bit enables/disables output of receive data full interrupts. Receive data empty interrupt is used as an interrupt factor to read the FIFO receive data register. Set this bit to 1 after setting set condition (SSISCR.RDFS) of Receive data empty interrupt. See **Figure 27.26** for the generation timing of a receive data full interrupt.

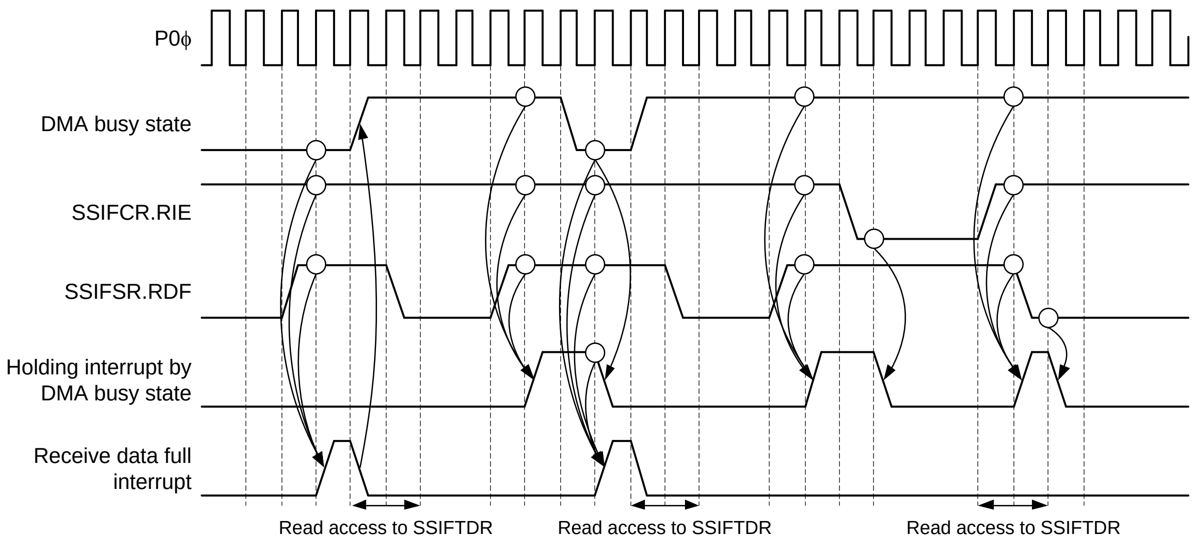


Figure 27.26 Generation timing of receive data empty interrupt

TFRST Bit

This bit sets a software reset of the transmit FIFO data register (SSIFTDR). Writing 1 to this bit initializes the internal state related to SSIFTDR. See **Table 27.8** for the registers to which this software reset is applied. After a reset by writing 1, write 0 to clear the reset condition because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.

This bit is a bit of SSIRST for reset. Software reset by SSIRST becomes and writing this bit becomes invalid because of priority.

Table 27.8 Bits Initialized by Software Reset of the SSIFCR.TFRST Bit

Symbol	Address (Base+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H'00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H'04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H'10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H'14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H'18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H'1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	H'20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H'24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

RFRST Bit

This bit sets a software reset of the receive FIFO data register (SSIFRDR). Writing 1 to this bit initializes the internal state related to SSIFRDR. After a reset by writing 1, write 0 to clear the reset condition because this bit is not automatically cleared to 0. After writing 0 to this bit, make sure that the bit becomes 0 before the next procedure.

This bit is a bit of SSIRST for reset. Software reset by SSIRST becomes and writing this bit becomes invalid because of priority.

Table 27.9 Bits Initialized by Software Reset of the SSIFCR.TFRST Bit

			+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Symbol	Address (Base+)		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H'00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H'04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H'10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H'14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H'18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H'1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR	H'20	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H'24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				

27.4.1.4 IFO Status Register (SSIFSR)

This register is configured with status flags that indicate the status of the transmit FIFO data register and the receive FIFO data register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W0

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	H'0	R	Reserved. Write 0. The read value is 0.
29 to 24	TDC[5:0]	H'00	R	Number of Transmit FIFO Data Indication Flag
23 to 17	—	H'00	R	Reserved. Write 0. The read value is 0.
16	TDE	1	RW0	Transmit Data Empty Flag 0: The free space of SSIFTDR is not more than the value of SSISCR.TDES 1: The free space of SSIFTDR is not less than the value of SSISCR.TDES plus one.
15, 14	—	H'0	R	Reserved. Write 0. The read value is 0.
13 to 8	RDC[4:0]	H'00	R	Number of Receive FIFO Data Indication Flag
7 to 1	—	H'00	R	Reserved. Write 0. The read value is 0.
0	RDF	0	RW0	Receive Data Full Flag 0: The size of received data in SSIFRDR is not more than the value of SSISCR.RDFS 1: The size of received data in SSIFRDR is not less than the value of SSISCR.RDFS plus one.

TDC[5:0] Bits

These bits indicate the number of valid data that are stored in the transmit FIFO data register (SSIFTDR). With this flag as H'0, there is no data to be transmitted. With H'10, there is no space to write data.

TDE Bit

This bit indicates that the transmit FIFO data register (SSIFTDR) has free space not less than the amount set with the SSIFCR.TTRG bit plus one. This flag is set by automatic determination but it must be cleared by register access.
[Priority order for setting and clearing]

Clearing is prioritized.*[1]

[Clearing condition]

Either of the following two:

(1) Writing 0 to this bit after reading this bit as 1.*[2]

(2) When writing data to SSIFTDR.

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this bit after reading this bit as 1 (**Figure 27.13**).

(2) When writing data to SSIFTDR.

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1) and transmit FIFO data register reset (SSIFCR.TFRST = 1). The software reset and transmit FIFO data register reset have priority over all the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following four conditions being met:
- Software reset (SSIFCR.SSIRST = 1)
- Resetting of the transmit FIFO data register (SSIFCR.TFRST = 1)
- Completion of writing 0 to this flag after having read it as 1.
- When writing data to SSIFTDR.

[Setting condition]

SSIFTDR has free space not less than the amount set with the SSIFCR.TTRG bit plus one.

[Setting timing]

While operating on P0ϕ, SSIFTDR is found to have free space not less than “size set in the SSISCR.TDES bits + 1.”

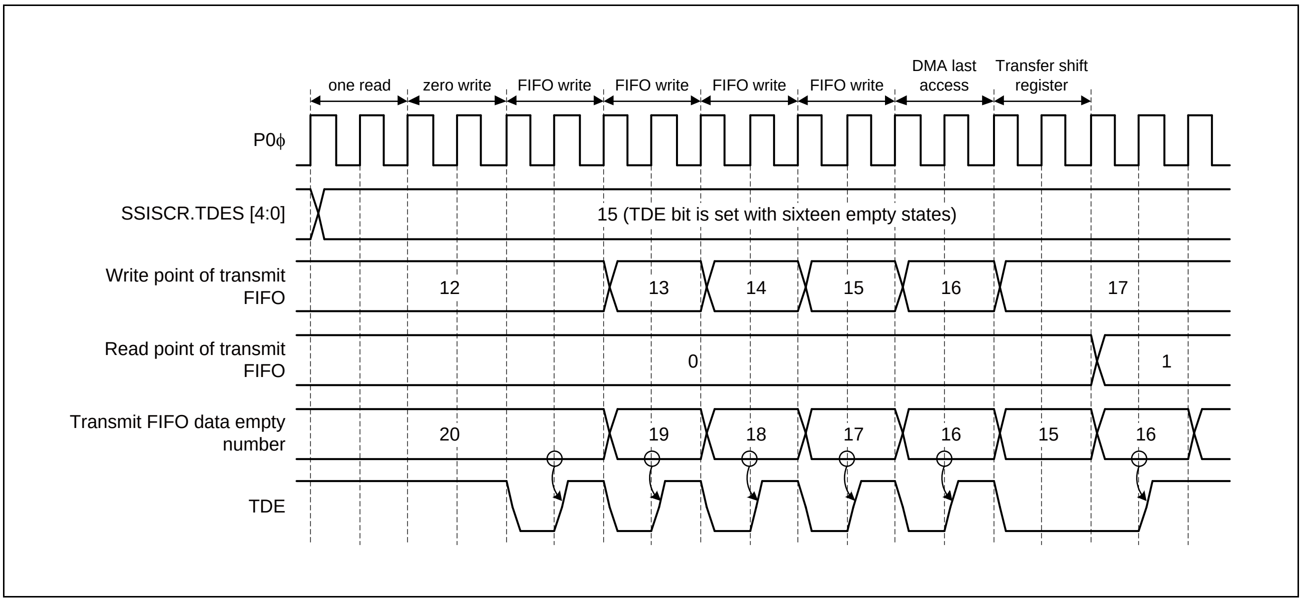


Figure 27.27 Set timing and clear timing of TDE

RDC[5:0] Bits

These bits indicate the number of valid data that are stored in the receive FIFO data register (SSIFRDR). With this flag as H’0, there is no received data. With H’20, the register is filled with received data and there is no free space.

RDF Bit

This bit indicates that the receive FIFO data register (SSIFRDR) has unread received data not less than the amount set with the SSISCR.RDFS bit plus one. This flag is set by automatic determination but it must be cleared by register access.

[Priority order for setting and clearing]

Clearing is prioritized.*1

[Clearing condition]

Either of the following two:

(1) Writing 0 to this bit after reading this bit as 1.*2

(2) When reading data from SSIFRDR.

[Clearing timing]

Timing according to the above-mentioned clear condition.

(1) At completion of writing 0 to this bit after reading this bit as 1 (**Figure 27.13**).

(2) After reading data from SSIFRDR.

Note 1. This bit is also cleared by a software reset (SSIFCR.SSIRST = 1) and receive FIFO data register reset (SSIFCR.RFRST = 1). Clearing conditions available for these bits are the software reset and receive FIFO data register reset as well as the clearing conditions described above.

Note 2. The state of reading this bit as one is cleared by any of the following four conditions being met:

- Software reset (SSIFCR.SSIRST = 1)
- Receive FIFO data register reset (SSIFCR.RFRST = 1)
- Completion of writing 0 to this flag after having read it as 1.
- When reading data from SSIFRDR.

[Setting condition]

SSIFRDR has data not less than the amount set with the SSISCR.RDFS bit plus one.

[Setting timing]

At completion of transfer from the shift register that results in SSIFRDR having data not less than the amount set with the SSISCR.RDFS bit plus one.

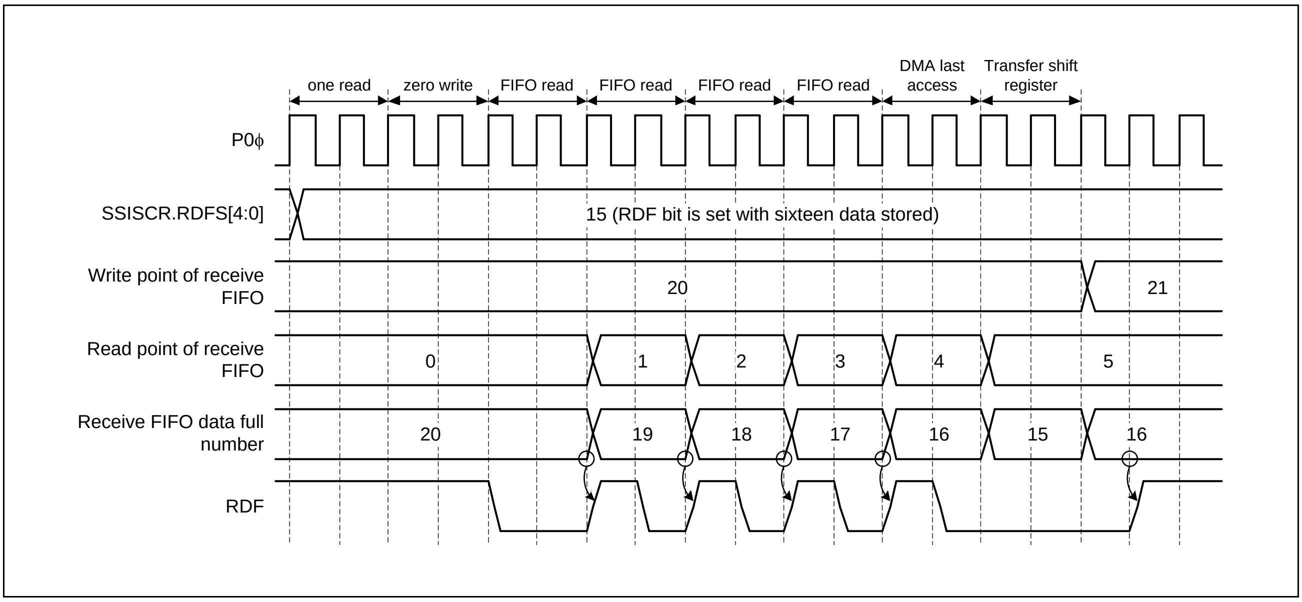


Figure 27.28 Set timing and clear timing of RDF

27.4.1.5 Transmit FIFO Data Register (SSIFTDR)

This is a 32-bit writable register. 0 is returned when this register is read. This register stores data to be serially transmitted.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSIFTDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSIFTDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SSIFTDR[31:0]	H’00000000	W	Transmit FIFO data

To use this register for transmission, set DMA operation handling of a transmit data empty interrupt as writing to this register. Determine the access size to this register according to the data word length to be communicated (**Table 27.10**).

Table 27.10 Register Access Restriction to FIFOs

		Access Size		
SSICR.DWL[2:0]	Data Word Length	Byte	Halfword	Word
000b	8	✓	—	—
001b	16	—	✓	—
010b	18	—	—	✓
011b	20	—	—	✓
100b	22	—	—	✓
101b	24	—	—	✓
110b	32	—	—	✓
111b	Setting prohibited	—	—	—

Figure 27.29 shows register access to the transmit FIFO data register.

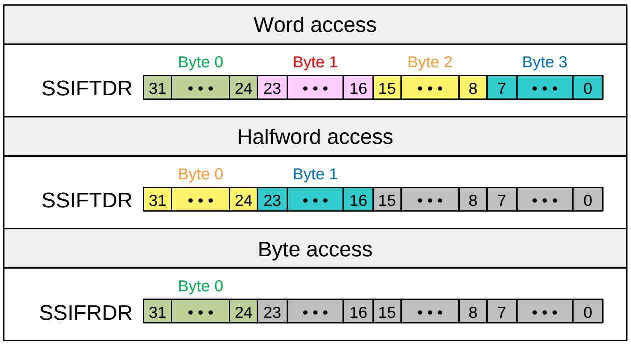


Figure 27.29 Example of Register Access to the Transmit FIFO Data Register

Figure 27.30 shows the configurations and operation examples of the transmit FIFO data register and transmit shift register. The configurations are for storing data to FIFO and not related with communication.

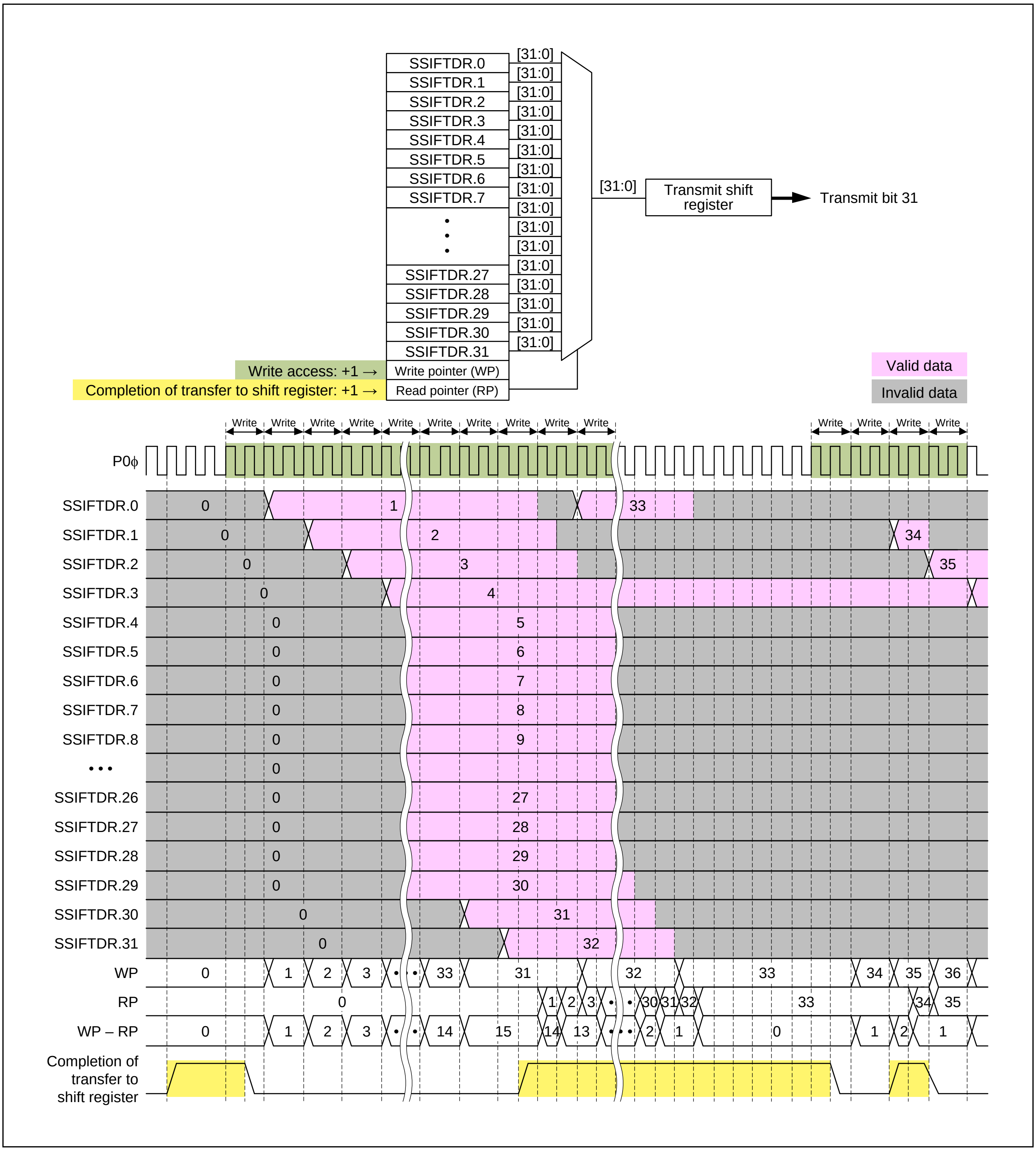


Figure 27.30 Configuration of the Transmit FIFO Data Register and Transmit Shift Register, and FIFO Operation Example

27.4.1.6 Receive FIFO Data Register (SSIFRDR)

This is a readable 32-bit register. This register stores received serial data.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSIFRDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSIFRDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SSIFRDR[31:0]	H’00000000	R	Receive FIFO data

To use this register for reception, set DMA operation handling of a receive data full interrupt as reading from this register. Determine the access size to this register according to the data word length to be communicated (**Table 27.10**). Register access to the receive FIFO data register is same as for the transmit FIFO data register (**Figure 27.29**).

Figure 27.31 shows the configurations and operation examples of the receive FIFO data register and receive shift register.

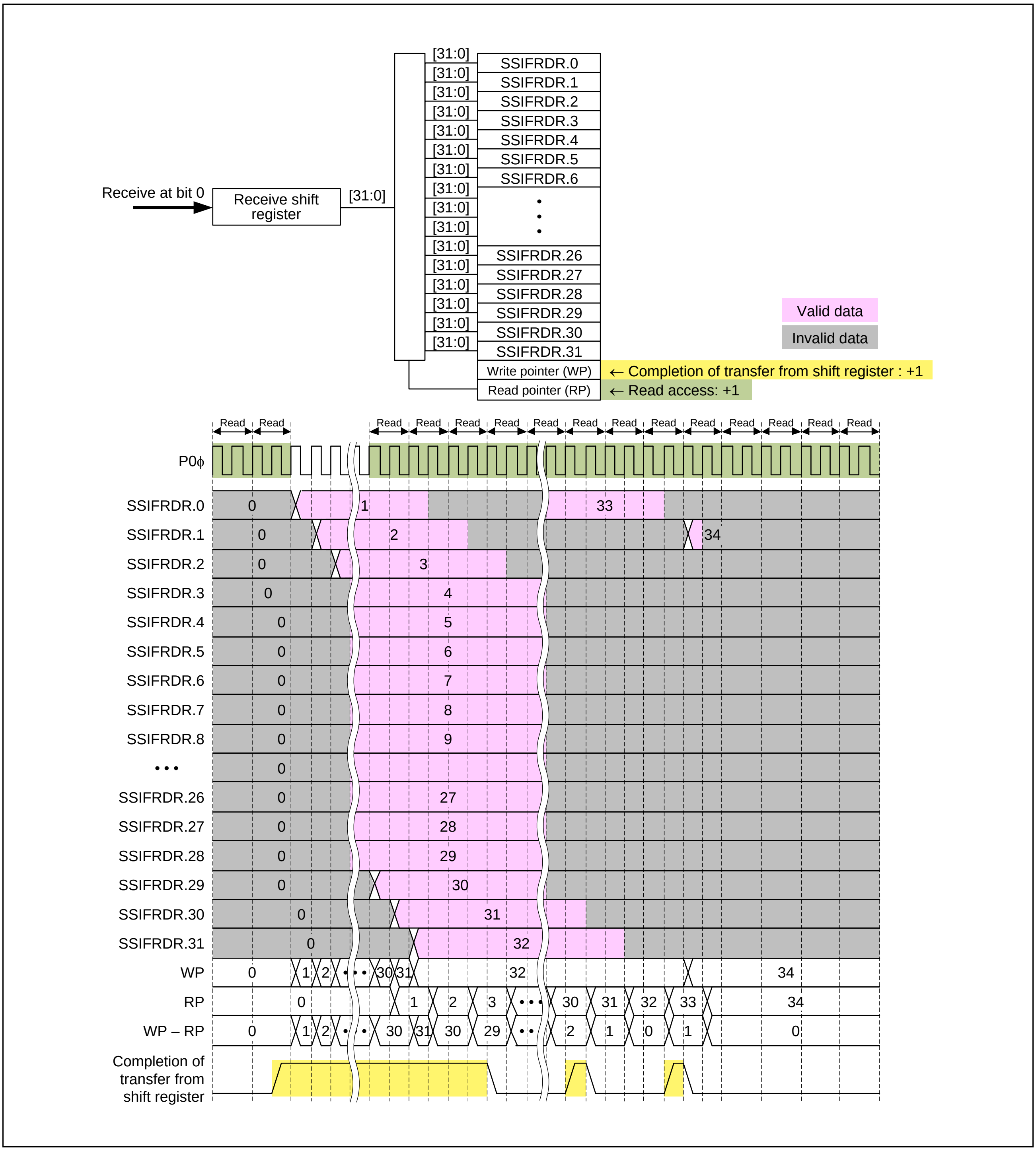


Figure 27.31 Configuration of the Receive FIFO Data Register and Receive Shift Register, and FIFO Operation Example

27.4.1.7 Audio Format Register (SSIOFR)

This is a readable/writable 32-bit register. It sets an audio format including the communication format, LRCK/FS continuation mode, and BCK output stop.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BCKSTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	H'000000	R	Reserved. Write 0. The read value is 0.
9	BCKASTP	0	RW	BCK Output Stop Enable[*1,*2] 0: Enables output of BCK 1: Disables output of BCK
8	LRCONT	0	RW	LRCK/FS Continuation Enable[*1,*2] 0: Disables LRCK continuation 1: Enables LRCK continuation
7 to 2	—	H'00	R	Reserved. Write 0. The read value is 0.
1, 0	OMOD[1:0]	0	RW	Audio Format Select[*3,*4] OMOD[1:0]: Format 00: I^{2}S format 01: TDM format 10: Monaural format 11: Setting prohibited

Note 1. This bit is valid only in master-mode communication (SSICR.MST = 1). The setting is invalid in slave-mode communication (SSICR.MST = 0).

Note 2. BCKASTP=1 and LRCONT=1 is prohibitions.

Note 3. Writing in this bit is a prohibition when SSIF-2 is communication state (SSISR.IIRQ=0). Operation is not guaranteed when rewriting it.

Note 4. Use the communication format setting that can be communicated when the format of the communication of the opposing device is compatible format of SSIF-2.

BCKASTP Bit

This bit turns on or off the function to output BCK to the SSIBCK pin according to the communication shown in **Figure 27.32** and **Figure 27.33** in master-mode communication (SSICR.MST = 1). Set this bit after setting completing the communication format.

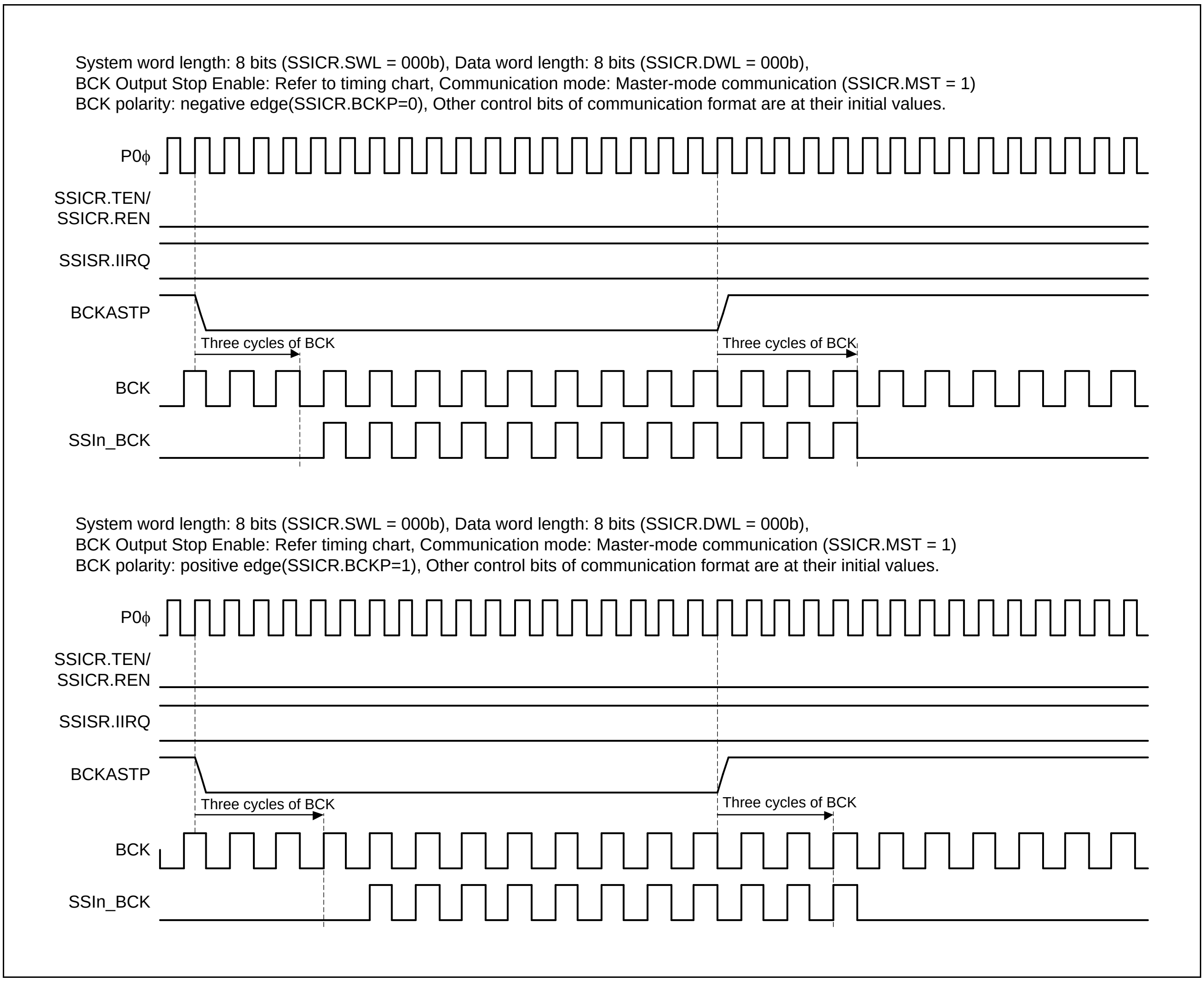
This bit must defend the following usage. Start communication with BCKASTP = 0, and set 1 to BCKASTP while communicating. As a result, the bit clock output to the terminal SSIBCK stops by the automatic operation when the communication stop is done. When you will restart the communication, set 0 to BCKASTP with idle state (SSICR.IIRQ=1) and the AUDIO_MCK supply state (SSIFCR.AUCKE = 1)

When master communication (SSICR.MST=1) and idle state (SSICR.IIRQ=1).

Table 27.11 BCKASTP state, and Output state of SSIBCK pin.

BCKASTP bit	output state of SSIBCK pin
0	Enable
1	Disable

Note: When the opposing device who is the slave needs the clock of the terminal SSIBCK before the communication operates, it is not possible to use it. Use it to stop clocking after the communication ends (**Figure 27.33**). Refer to **Figure 27.32** for timing that the function becomes effective.


System word length: 8 bits (SSICR.SWL = 000b), Data word length: 8 bits (SSICR.DWL = 000b),
BCK Output Stop Enable: Refer to timing chart, Communication mode: Master-mode communication (SSICR.MST = 1)
BCK polarity: negative edge(SSICR.BCKP=0), Other control bits of communication format are at their initial values.

System word length: 8 bits (SSICR.SWL = 000b), Data word length: 8 bits (SSICR.DWL = 000b),
BCK Output Stop Enable: Refer timing chart, Communication mode: Master-mode communication (SSICR.MST = 1)
BCK polarity: positive edge(SSICR.BCKP=1), Other control bits of communication format are at their initial values.

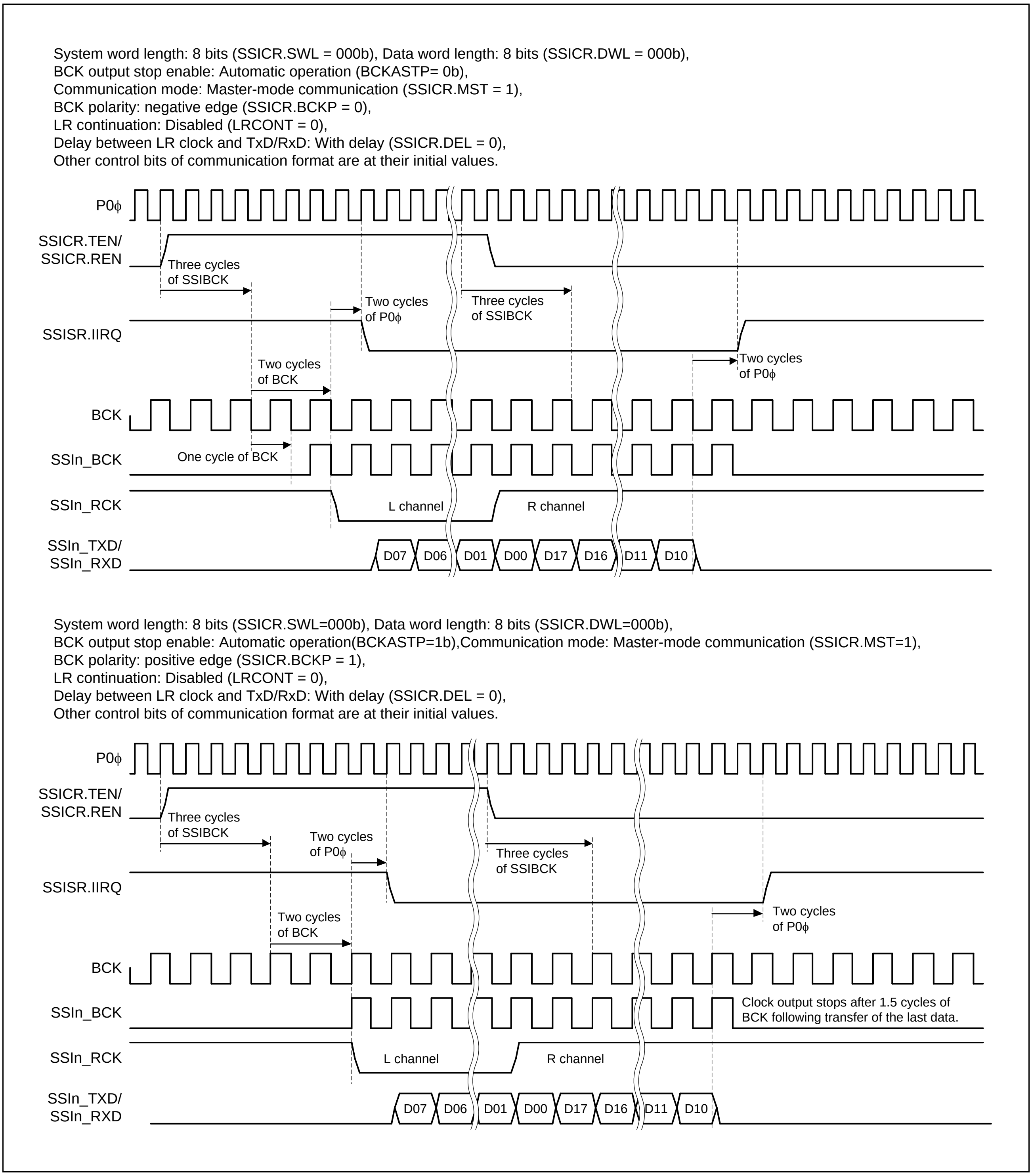
Figure 27.32 Example Operation of the BCKASTP Bit (in idle state)

When master communication (SSICR.MST=1) and the BCK output automatic operation stop function (BCKASTP = 1) (**Figure 27.33**).

Details of the BCK output to SSIBCK pin are as follows.

- Output start timing: To generate an effective edge in timing that LRCK/FS is changed into the valid value, BCK is output.
- Output stop timing: Frame boundary from 1 to 1.5 clock

Refer to the timing chart of **Figure 27.33** for detailed timing.



System word length: 8 bits (SSICR.SWL = 000b), Data word length: 8 bits (SSICR.DWL = 000b),
BCK output stop enable: Automatic operation (BCKASTP= 0b),
Communication mode: Master-mode communication (SSICR.MST = 1),
BCK polarity: negative edge (SSICR.BCKP = 0),
LR continuation: Disabled (LRCONT = 0),
Delay between LR clock and TxD/RxD: With delay (SSICR.DEL = 0),
Other control bits of communication format are at their initial values.

System word length: 8 bits (SSICR.SWL=000b), Data word length: 8 bits (SSICR.DWL=000b),
BCK output stop enable: Automatic operation(BCKASTP=1b),Communication mode: Master-mode communication (SSICR.MST=1),
BCK polarity: positive edge (SSICR.BCKP = 1),
LR continuation: Disabled (LRCONT = 0),
Delay between LR clock and TxD/RxD: With delay (SSICR.DEL = 0),
Other control bits of communication format are at their initial values.

Figure 27.33 Example Operation of the BCKASTP Bit (Communication behavior in BCKASTP=1)

LRCONT Bit

This bit enables/disables the output from the SSILRCK/SSIFS pin when SSIF-2 is master communication (SSICR.MST=1) and idle state (SSISR.IIRQ=1). Writing 1 to this bit (enables LRCK/FS continuation) in master mode (SSICR.MST = 1) enables continuation of the output from the SSILRCK/SSIFS pins even when idle state.

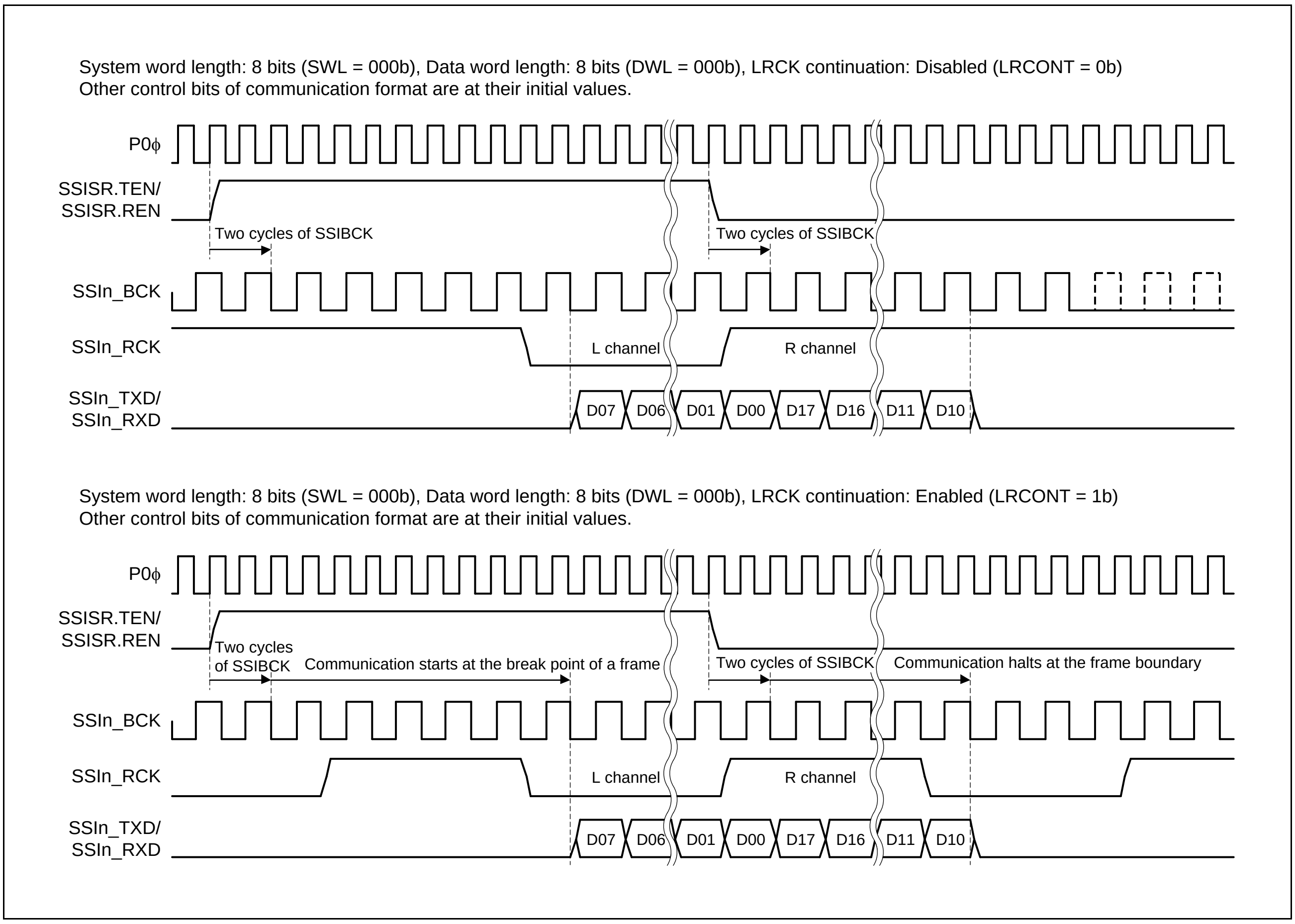


Figure 27.34 Operation Example of LRCK/FS Continuation (in idle state)

OMOD[1:0] Bits

These bits set an audio format.

Rewrite this bit when the LR clock supply to the terminal SSILRCK stops. Refer to the LRCONT bit details explanation in **Section 27.4.1.7, Audio Format Register (SSIOFR)** for the output operation of the LR clock.

27.4.1.8 Status Control Register (SSISCR)

This is a readable/writable 32-bit register. It sets the operation of the TDE and RDF flags.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TDES[4:0]					—	—	—	RDFS[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	H'00000	R	Reserved. Write 0. The read value is 0.
12 to 8	TDES[4:0]	H'00	RW	TDE Setting Condition Select*[1] TDES[4:0]: Operation 00000b: SSIFTDR has one stage or more free space 00001b: SSIFTDR has two stages or more free space (snip) 11110b: SSIFTDR has thirty-one stages or more free space 11111b: SSIFTDR has thirty-two stages or more free space
7 to 5	—	H'0	R	Reserved. Write 0. The read value is 0.
4 to 0	RDFS[4:0]	H'00	RW	RDF Setting Condition Select*[1] RDFS[4:0]: Operation 00000b: SSIFRDR has one stage or more data size 00001b: SSIFRDR has two stages or more data size (snip) 01110b: SSIFRDR has thirty-one or more data size 01111b: SSIFRDR has thirty-two stages or more data size

Note 1. Writing to these bits while SSIF-2 is in a communication state (SSISR.IIRQ = 0) is prohibited. If written, the operation performed immediately after writing is not guaranteed.

TDES[4:0] Bits

These bits set the setting condition of the transmit data empty flag (TDE). Set these bits in byte units.

RDFS[4:0] Bits

These bits set the setting condition of the receive data full flag (RDF). Set these bits in byte units.

27.4.2 Communication Formats

SSIF-2 supports three communication formats (**Table 27.12**).

Table 27.12 Supported Communication Formats

Communication Format	SSIOFR.OMOD[1:0]
I^{2}S format	00
TDM format	01
Monaural format	10

The following figure shows the serial data configuration common to communication formats. Serial data is determined according to the system word length (SSICR.SWL[2:0]) and the data word length (SSICR.DWL[2:0]). When the system word length is longer than the data word length, padding bits are transferred (**Figure 27.35**).

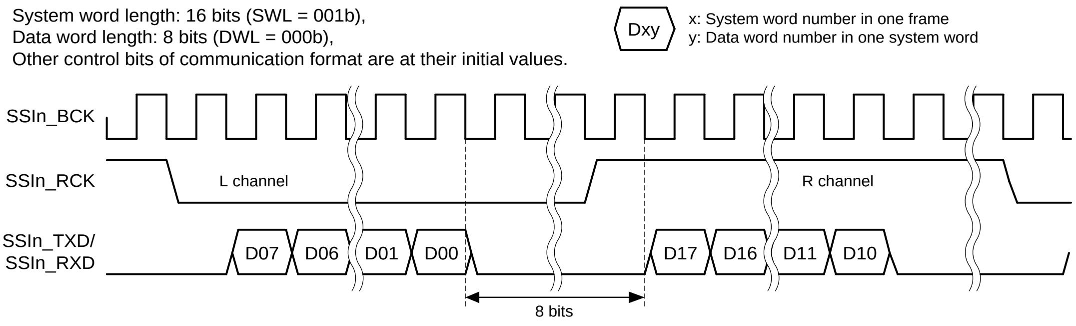


Figure 27.35 Example of Padding Bit Transfer (I^{2}S Format: System Word Length > Data Word Length)

Table 27.13 lists the number of padding bits to be transferred with each combination of system word length (SSICR.SWL[2:0]) and data word length (SSICR.DWL[2:0]). "-" indicates that the setting is prohibited.

Table 27.13 Number of Padding Bits

	SSICR.DWL[2:0]	000b	001b	010b	011b	100b	101b	110b	111b
SSICR.SWL[2:0]	System Word Length	8	16	18	20	22	24	32	Set prohibition
000b	8	0	—	—	—	—	—	—	—
001b	16	8	0	—	—	—	—	—	—
010b	24	16	8	6	4	2	0	—	—
011b	32	24	16	14	12	10	8	0	—
100b	48	40	32	30	28	26	24	16	—
101b	64	56	48	46	44	42	40	32	—
110b	128	120	112	110	108	106	104	96	—
111b	256	248	240	238	236	234	232	224	—

27.4.2.1 I^{2}S Format

The I²S format is a communication format used for connection with I²S-compatible serial devices. With this format setting (SSIOFR.OMOD[1:0] = 00b), one frame is configured with two system words, one for the channel L and the other for channel R. The SSILRCK/SSIFS signals are at a low level for the channel L and at a high level for the channel R. Set the polarity of the signals with the SSICR.LRCKP bit. **Figure 27.36** shows the I^{2}S format without padding. See **Figure 27.35** for the format with padding.

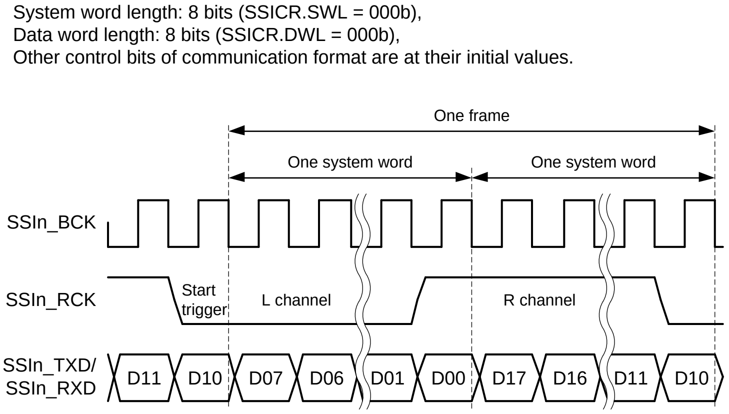


Figure 27.36 I^{2}S Format (Without Padding: System Word Length = Data Word Length)

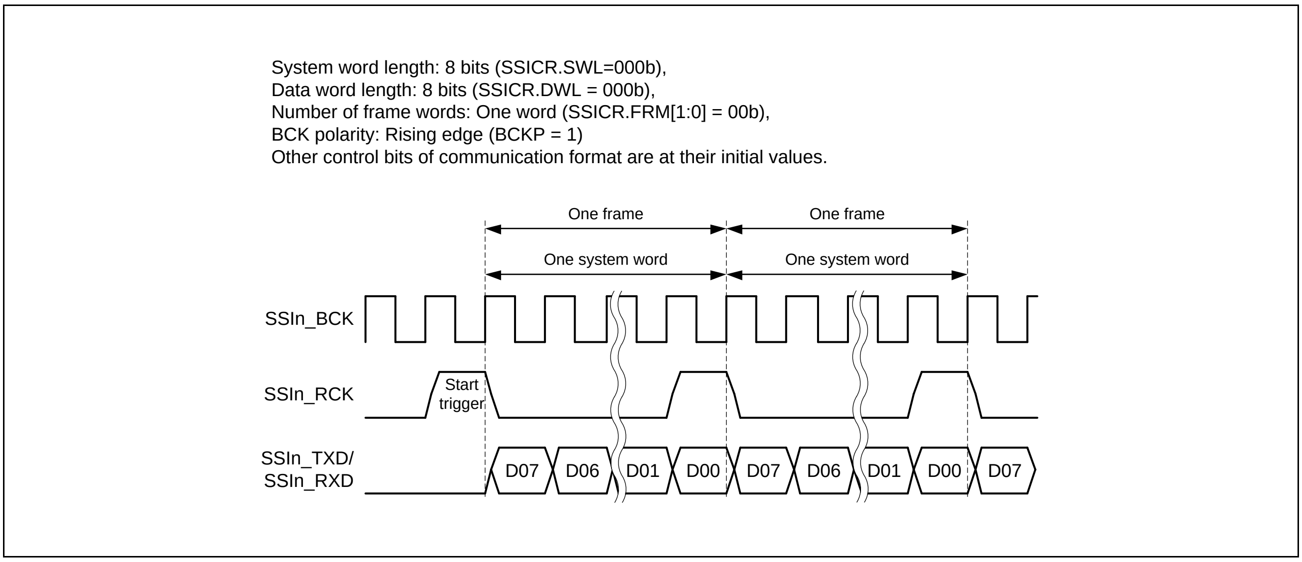
For the state of external pins when SSIF-2 is in the idle state, see **Section 27.5.1.1, Idle State**.

NOTE

SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/ SSIFS only as a communication start trigger.

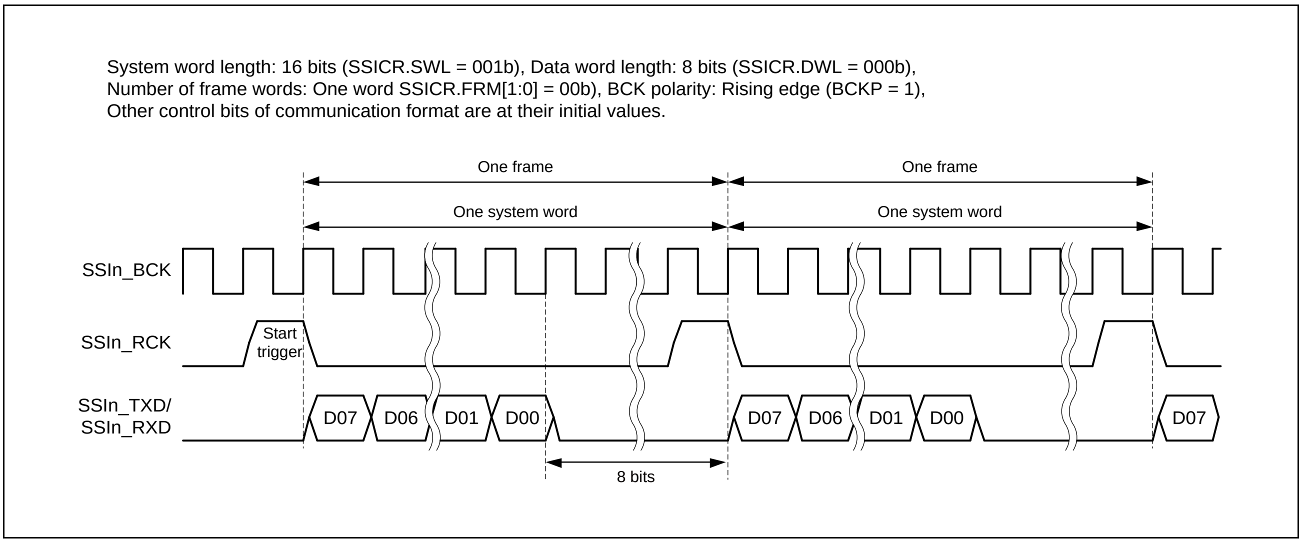
27.4.2.2 Monaural Format

The monaural format is a communication format used for connection with monaural-compatible serial devices. With this format setting (SSIOFR.OMOD[1:0] = 10b), one frame is configured with the number of system words set with the SSICR.FLM[1:0] bits (**Figure 27.37**). A rising of the SSILRCK/SSIFS signal means a start trigger. **Figure 27.37** and **Figure 27.38** respectively show the monaural formats without and with padding.



System word length: 8 bits (SSICR.SWL=000b),
Data word length: 8 bits (SSICR.DWL = 000b),
Number of frame words: One word (SSICR.FRM[1:0] = 00b),
BCK polarity: Rising edge (BCKP = 1)
Other control bits of communication format are at their initial values.

Figure 27.37 Short Frame in Monaural Format (Without Padding: System Word Length = Data Word Length)



System word length: 16 bits (SSICR.SWL = 001b), Data word length: 8 bits (SSICR.DWL = 000b),
Number of frame words: One word SSICR.FRM[1:0] = 00b), BCK polarity: Rising edge (BCKP = 1),
Other control bits of communication format are at their initial values.

Figure 27.38 Short Frame in Monaural Format (With Padding: System Word Length > Data Word Length)

The monaural formats supported by SSIF-2 consist of short frames and long frames. See **Section 27.4.2.2(1), Short Frame** and **Section 27.4.2.2(2), Long Frame** for the difference between these two frames.

For the state of external pins state when SSIF-2 is in the idle state, see **Section 27.5.1.1, Idle State**.

NOTE

SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

(1) Short Frame

With the short frame (SSICR.DEL = 0), the SSILRCK/SSIFS signal that indicates a start of serial data is at a high level for one cycle of SSIBCK. Data transfer starts at the falling edge of the signal (**Figure 27.37** and **Figure 27.38**).

(2) Long Frame

With the long frame (SSICR.DEL = 1), the SSILRCK/SSIFS signal that indicates a start of serial data is at a high level for two cycles of SSIBCK. Data transfer starts at the rising edge of the signal (**Figure 27.39**).

System word length: 8 bits (SSICR.SWL = 000b),
Data word length: 8 bits (SSICR.DWL= 000b),
Number of frame words: One word (SSICR.FRM[1:0] = 00b)
Delay of transfer data: No delay (SSICR.DEL = 1)
BCK polarity: Rising edge (BCKP = 1)
Other control bits of communication format are at their initial values.

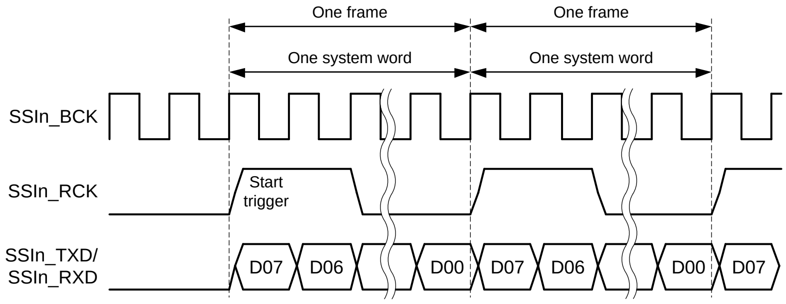


Figure 27.39 Long Frame in Monaural Format (Without Padding)

27.4.2.3 TDM Format

The TDM format is a communication format used for connection with TDM-compatible multi-channel devices. With this format setting (SSIOFR.OMOD[1:0] = 01b), one frame is configured with four to eight system words set with the SSICR.FRM[1:0] bits. With this format, the SSILRCK/SSIFS signal is at a high level for the first one system word and at a low level for the rest. The pulse generated on the SSILRCK/SSIFS signal is defined as the SYNC pulse and its rising edge means a start of one frame. **Figure 27.40** and **Figure 27.41** respectively show the TDM formats with and without padding.

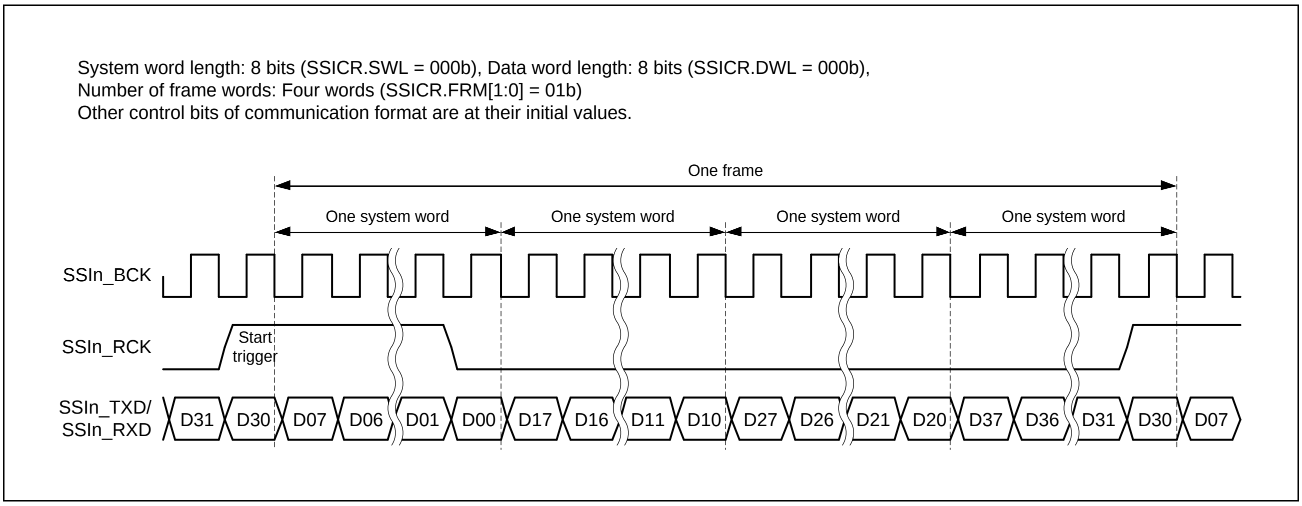

System word length: 8 bits (SSICR.SWL = 000b), Data word length: 8 bits (SSICR.DWL = 000b),
Number of frame words: Four words (SSICR.FRM[1:0] = 01b)
Other control bits of communication format are at their initial values.

Figure 27.40 TDM Format (Without Padding: System Word Length = Data Word Length)

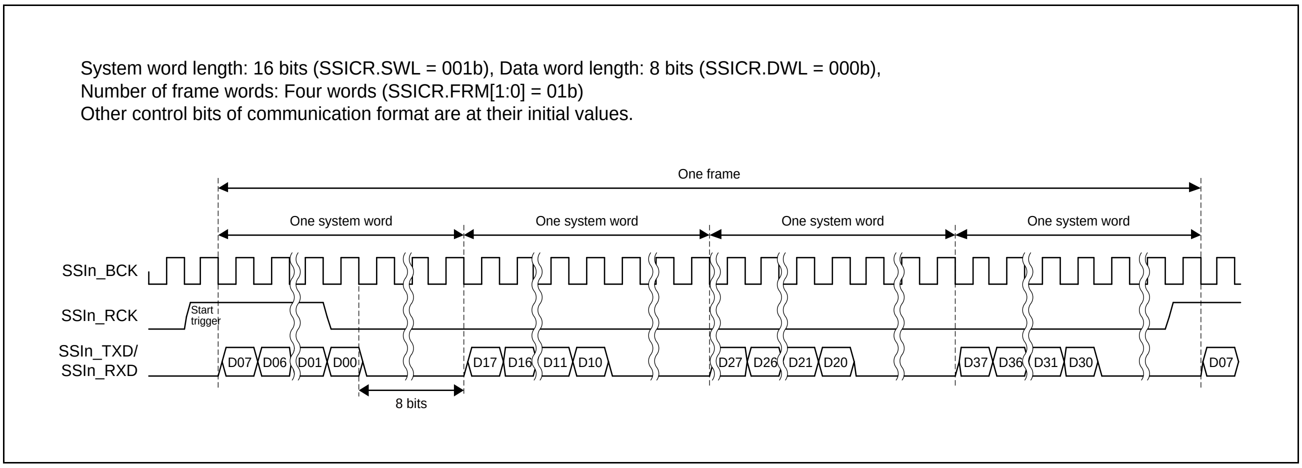

System word length: 16 bits (SSICR.SWL = 001b), Data word length: 8 bits (SSICR.DWL = 000b),
Number of frame words: Four words (SSICR.FRM[1:0] = 01b)
Other control bits of communication format are at their initial values.

Figure 27.41 TDM Format (With Padding: System Word Length > Data Word Length)

For the state of external pins when SSIF-2 is in the idle state, see **Section 27.5.1.1, Idle State**.

NOTE

SSIF-2 has the terminal SSILRCK/SSIFS that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSILRCK/SSIFS only as a communication start trigger.

27.4.3 Communication Modes

SSIF-2 supports the following communication modes. **Table 27.15** lists the control bits that are not available with each communication mode. See **Section 27.4.3.1** to **Section 27.4.3.5** for details of these communication modes.

Table 27.14 Communication Modes

Communication Mode	SSICR.MST Bit	SSICR.REN Bit	SSICR.TEN Bit
Slave-mode transmission	0	0	1
Slave-mode reception	0	1	0
Slave-mode transmission and reception	0	1	1
Master-mode transmission	1	0	1
Master-mode reception	1	1	0
Master-mode transmission and reception	1	1	1

Table 27.15 Control Bits that Cannot be Used in Each Communication Mode

Communication Mode Control Bit	Slave-mode Reception	Slave-mode Transmission	Slave-mode Transmission and Reception	Master-mode Reception	Master-mode Transmission	Master-mode Transmission and Reception
SSICR.CKS	Invalid	Invalid	Invalid	Available	Available	Available
SSICR.CKDV	Invalid	Invalid	Invalid	Available	Available	Available
SSICR.MUEN	Invalid	Available	Available	Invalid	Available	Available
SSICR.TEN	Invalid	Available	Available	Invalid	Available	Available
SSICR.REN	Available	Invalid	Available	Available	Invalid	Available
SSIFCR.AUCKEN	Invalid	Invalid	Invalid	Available	Available	Available
SSIFCR.BCKNCE	Available	Available*[1]	Available*[2]	Invalid	Invalid	Invalid
SSIFCR.LRCKNCE	Available	Available*[1]	Available*[2]	Invalid	Invalid	Invalid
SSIFCR.RXDNCE	Available	Invalid	Available	Invalid	Invalid	Invalid
SSIFCR.TIE	Invalid	Available	Available	Invalid	Available	Available
SSIFCR.RIE	Available	Invalid	Available	Available	Invalid	Available
SSIFCR.TFRST	Invalid	Available	Available	Invalid	Available	Available
SSIFCR.RFRST	Available	Invalid	Available	Available	Invalid	Available
SSIOFR.BCKASTP	Invalid	Invalid	Invalid	Available	Available	Available
SSIOFR.LRCONT	Invalid	Invalid	Invalid	Available	Available	Available
SSIOFR.OMOD	Available	Available	Available	Available	Available	Available
SSISCR.TDES	Invalid	Available	Available	Invalid	Available	Available
SSISCR.RDFS	Available	Invalid	Available	Available	Invalid	Available

Note 1. Disabled when SSICR.DEL = ‘1’

Note 2. Note 2. Disabled during transmission (SSICR.TEN = ‘1’) when SSICR.DEL = ‘1’

27.4.3.1 Slave-mode Communication

SSIF-2 operates in slave mode with SSICR.MST = 0. Supply the SSIBCK signal and the SSILRCK/SSIFS signal used for the serial data communication from external devices. If these signals do not match the communication format set for SSIF-2, operation is not guaranteed.

27.4.3.2 Master-mode Communication

SSIF-2 operates in master mode with SSICR.MST = 1. The SSIBCK signal and SSILRCK/SSIFS signal used for the serial data communication are internally generated from the audio clock. These signals use the format according to the setting of SSIF-2. If the communication format of another device working as the slave device does not match the communication format set for SSIF-2, operation is not guaranteed.

27.4.3.3 Transmission

SSIF-2 transmits serial data to opposing device when SSICR.TEN = 1 and SSICR.REN = 0. If the communication format of the opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

27.4.3.4 Reception

SSIF-2 receives serial data from opposing device when SSICR.TEN = 0 and SSICR.REN = 1. If the communication format of the opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

27.4.3.5 Transmission and Reception

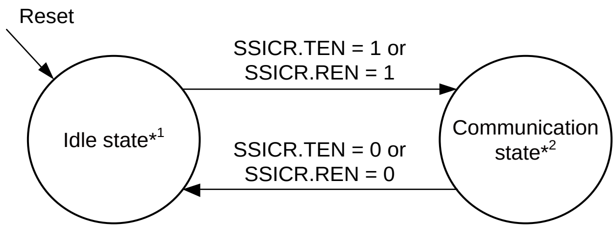
SSIF-2 transmits and receives serial data to and from opposing device when SSICR.TEN = 1, SSICR.REN = 1. If the communication format of opposing device does not match the communication format set for SSIF-2, operation is not guaranteed.

27.5 Operation

27.5.1 Operational State

SSIF-2 has the following two main operation states (**Figure 27.42**).

- Idle state (SSISR.IIRQ = 1)
- Communication state (SSISR.IIRQ = 0)



Note 1. See **Section 27.5.1.1** for details of the idle state.

Note 2. See **Section 27.5.1.2** for details of the communication state.

Figure 27.42 SSIF-2 State Transition

27.5.1.1 Idle State

In this state, communication of SSIF-2 is halted. However, when SSICR.MST = 1, output of BCK and LRCK/FS to external pins is enabled depending on the settings in the SSIOFR.BCKASTP bit and the SSIOFR.LRCONT bit (**Table 27.16**). This function is common to all formats.

Table 27.16 Output from External Pins in the Idle State

			Output from Pins		
SSICR.MST	SSIOFR.BCKASTP	SSIOFR.LRCONT	SSIBCK	SSILRCK/SSIFS	SSITxD
0	—	—	Stop	Stop	Stop
1	0	0	Supply	Stop	Stop
1	0	1	Supply	Supply	Stop
1	1	0	Stop	Stop	Stop
1	1	1	Stop	Supply	Stop

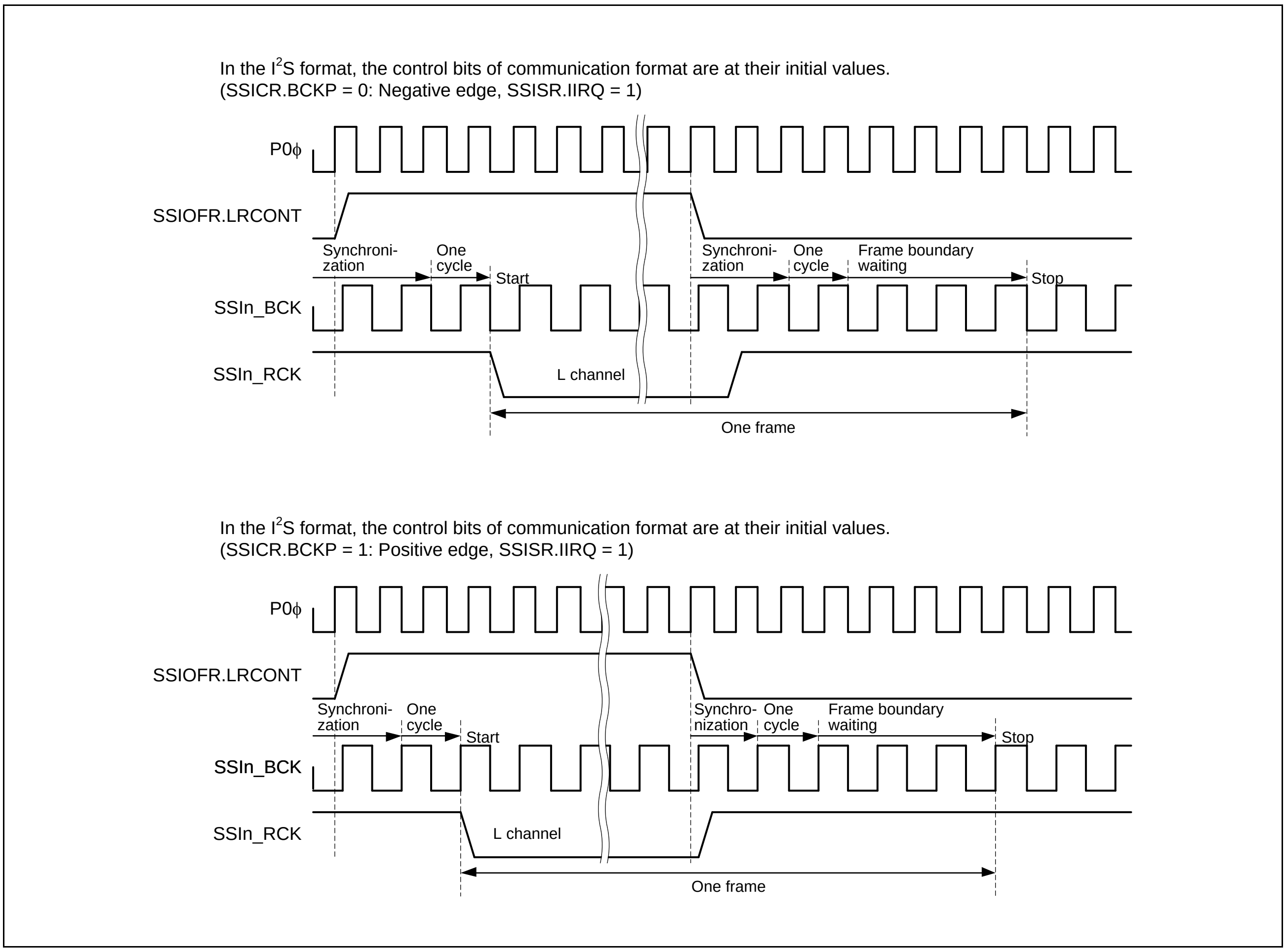


Figure 27.43 Example of LRCK/FS Continuation Release with SSIOFR.LRCONT

NOTE

To stop the output to the SSILRCK/SSIFS pin with SSIOFR.LRCONT when SSIF-2 is in the idle state in master- mode communication (SSICR.MST = 1), note the following: The output stops when 0 is written from 1 to SSIOFR.LRCONT. (**Figure 27.43**). Make sure that the remote device is not affected.

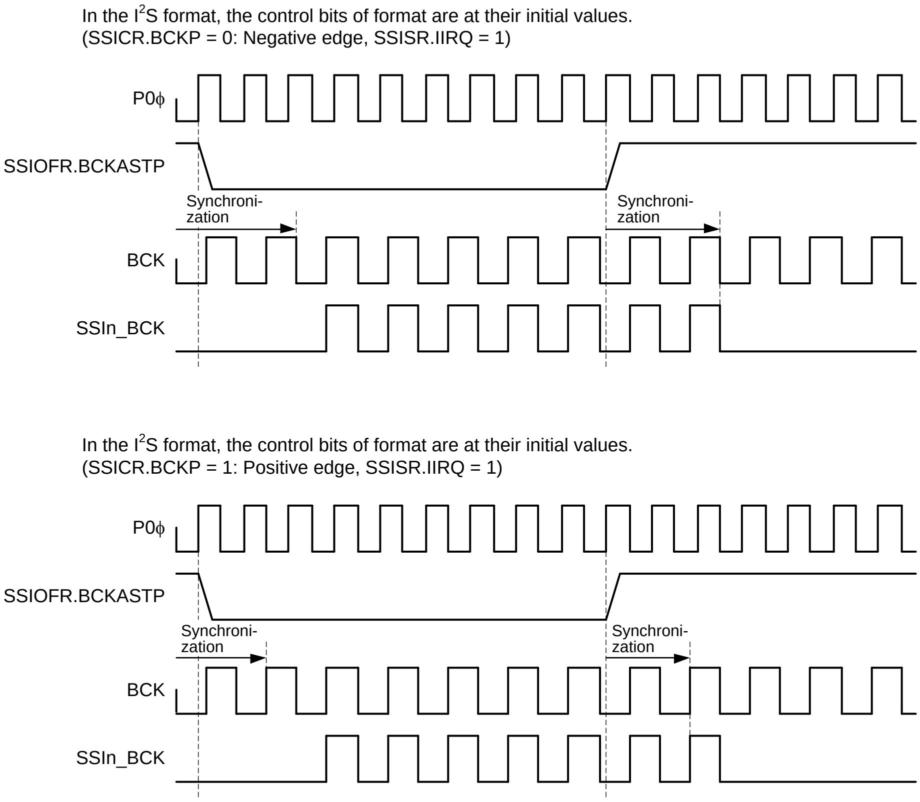


Figure 27.44 Example of Stopping SSIBCK with SSIOFR.BCKASTP

NOTE

To stop the output to the SSIBCK pin with SSIOFR.BCKASTP in master-mode communication (SSICR.MST = 1) and while SSIF-2 is in the idle state, note the following; By writing 0 to SSIOFR.BCKASTP while it is 1, the output stops immediately (**Figure 27.44**). So, make sure that the remote device is not affected.

27.5.1.2 Communication States

In this state, SSIF-2 is during communication. **Figure 27.45** shows transitions of communication states and **Table 27.17** lists the conditions for transition. If the transition condition is not satisfied, the state does not transit.

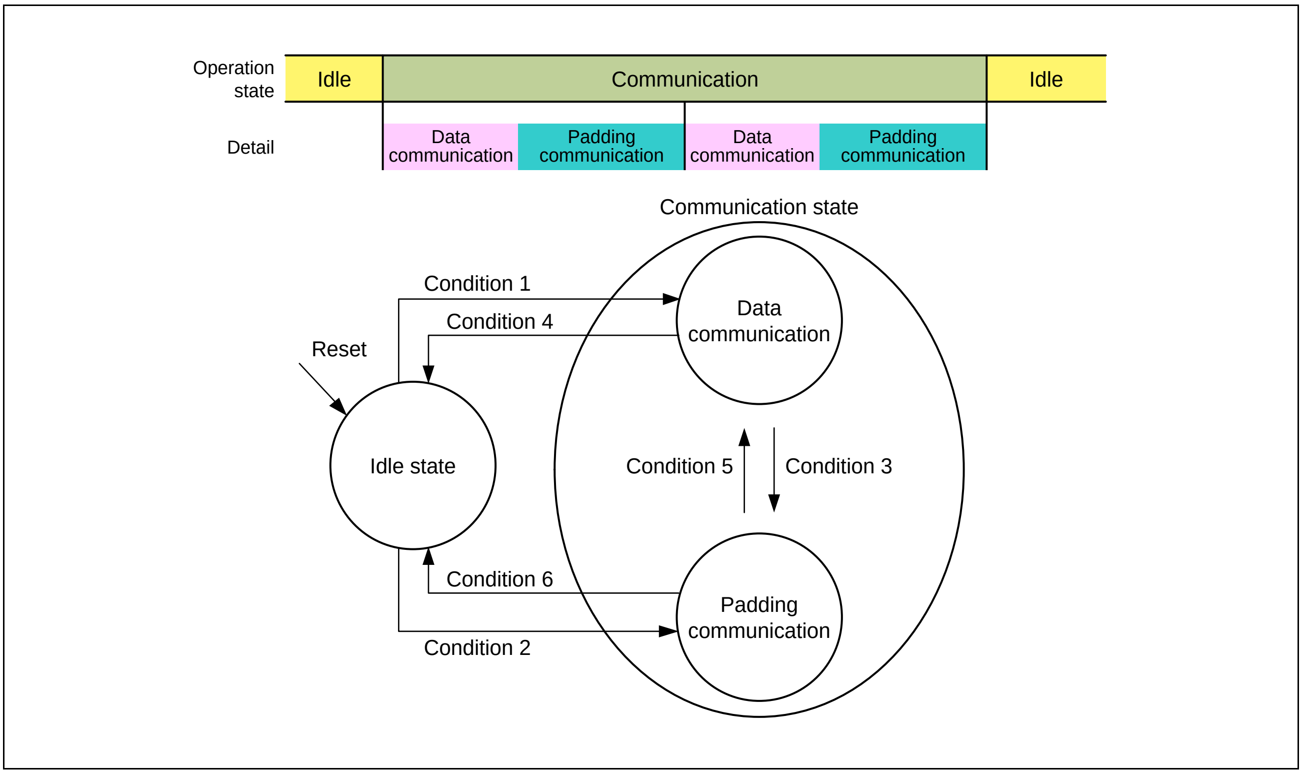


Figure 27.45 Communication State Transition

Table 27.17 Condition for Communication State Transition

Condition No.	Condition for Transition
1	Writing SSICR.TEN = 1 or SSICR.REN = 1 while SSICR.SDTA = 0 or in the setting without padding bits.
2	Writing SSICR.TEN = 1 or SSICR.REN = 1 while SSICR.SDTA = 1 and in the setting with padding bits.
3	The following all of three conditions are satisfied: ● SSICR.TEN = 1 or SSICR.REN = 1 ● In the setting without padding bits ● Transfer data words of the last data bit is completed.
4	The following all of two conditions are satisfied: ● SSICR.SDTA = 1 or without padding bits ● Transfer data words of the last data bit of one frame is completed while SSICR.TEN = 0 and SSICR.REN = 0
5	Transfer of the last padding bit is completed while SSICR.TEN = 1 or SSICR.REN = 1
6	The following all of two conditions are satisfied: ● SSICR.SDTA = 0 and with padding bits ● Transfer of the last data bit of one frame is completed while SSICR.TEN = 0 and SSICR.REN = 0.

See **Table 27.13** for the setting with/without padding bits.

(1) Data Communication State

In this state, SSIF-2 is during communication. Data of the data word length set with the SSICR.DWL[2:0] bits is transmitted, received, or transmitted and received.

State Transition in the Setting without Padding Bits

During communication (SSISR.IIRQ = 0), SSIF-2 is during data communication for all the time (**Figure 27.46**). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0), SSIF-2 transits to the idle state (**Figure 27.47**).

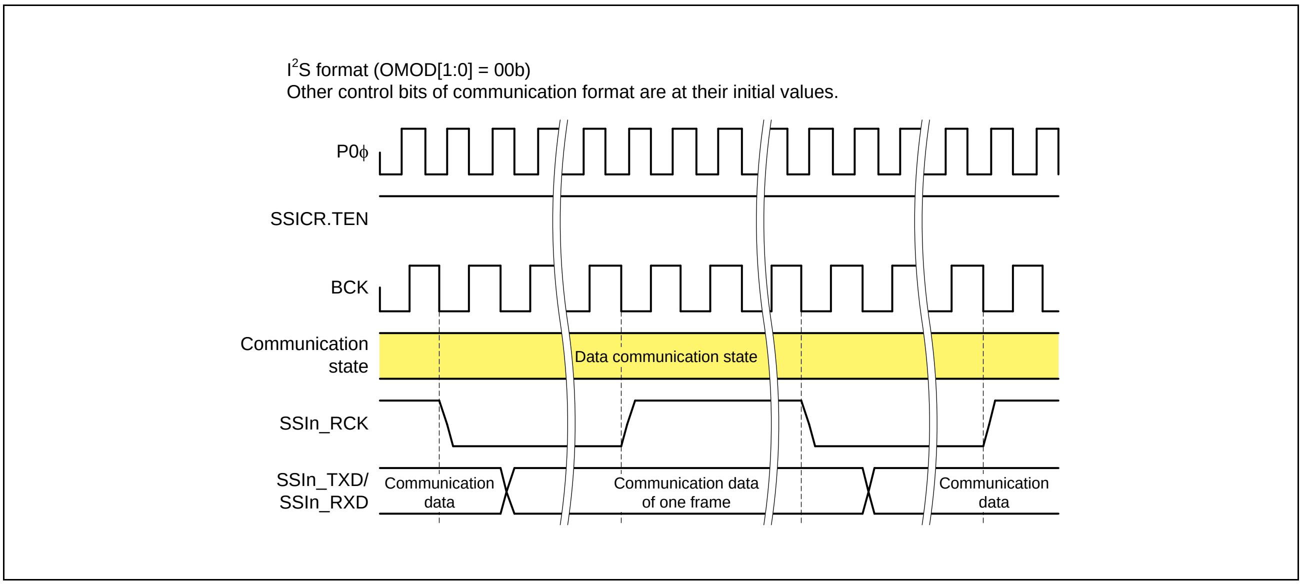


Figure 27.46 Continuation of the Data Communication State

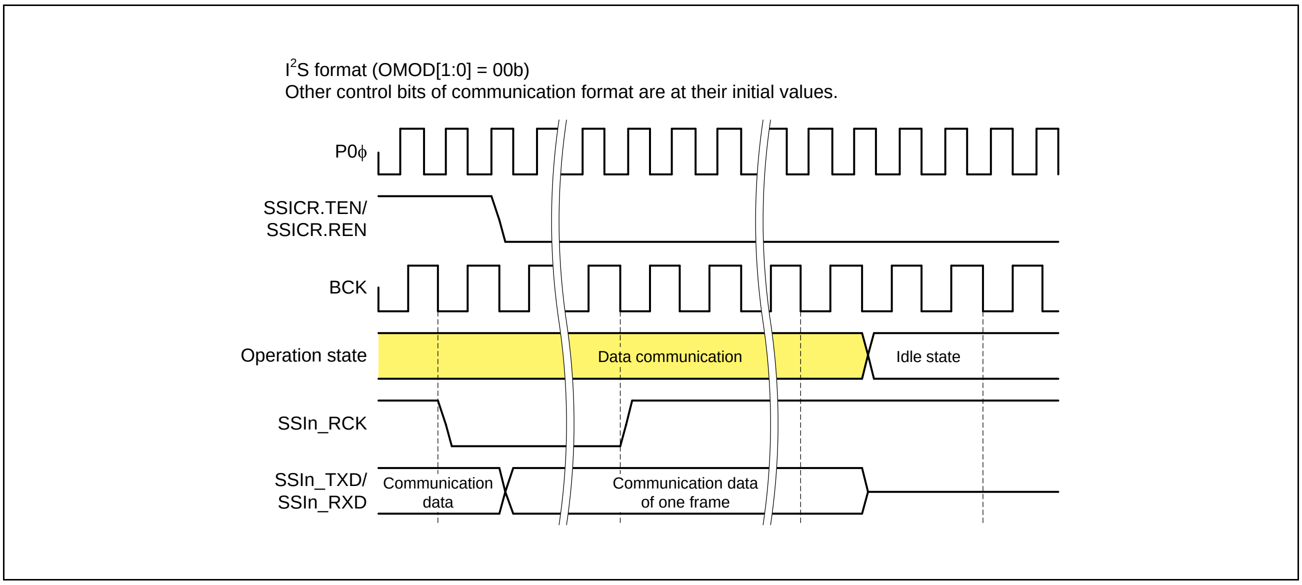


Figure 27.47 Halt from the Data Communication State (without Padding Bits)

State Transition in the Setting with Padding Bits

During communication (SSISR.IIRQ = 0), after completing transfer of the last bit of a data word, SSIF-2 transits to the padding communication state from data communication state (**Figure 27.48**). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0) while SSICR.SDTA = 1, SSIF-2 transits to the idle state from data communication state when the communication stop is done. (**Figure 27.49**).

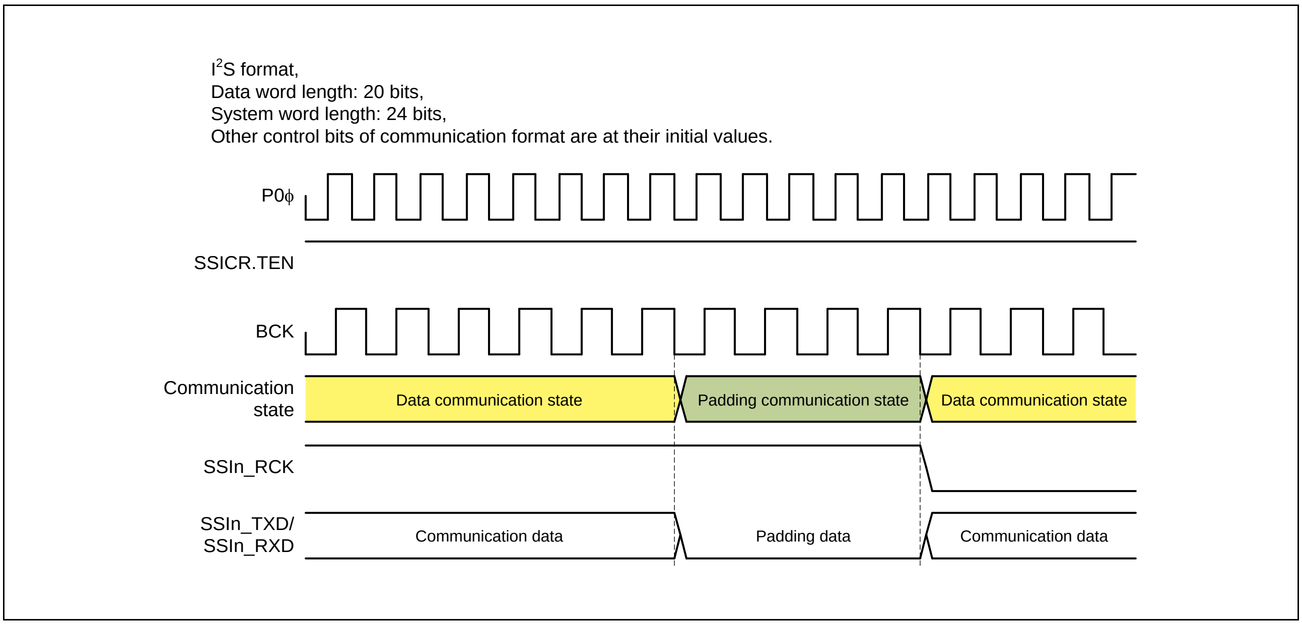


Figure 27.48 Transition from Data Communication to Padding Communication

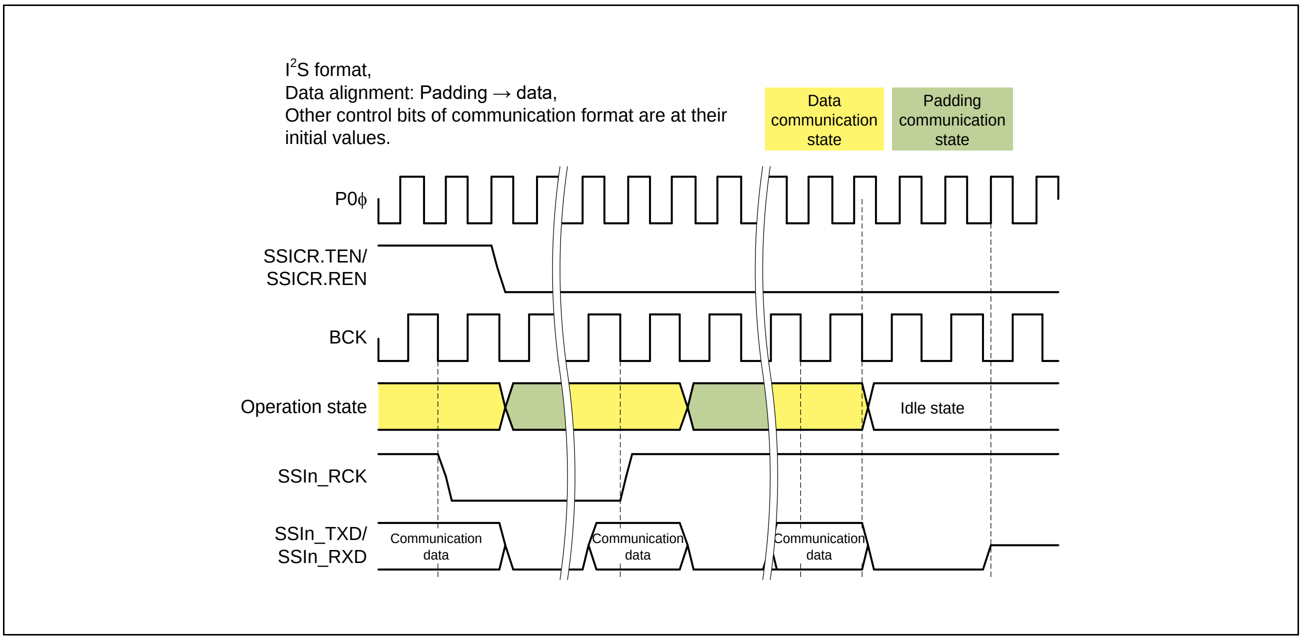


Figure 27.49 Halt from Data Communication (with Padding Bits)

(2) Padding Communication State

In this state, SSIF-2 is during communication. The padding bits set with the SSICR.SWL[2:0] bits and SSICR.DWL[2:0] bits are transmitted, received, or transmitted and received.

State Transition in the Setting with Padding Bits

During communication (SSISR.IIRQ = 0), after completing transfer of the last padding bit, SSIF-2 transits to data communication state (**Figure 27.48**). By disabling transmission and reception (SSICR.TEN = 0 and SSICR.REN = 0) while SSICR.SDTA = 0, SSIF-2 transits to the idle state from padding communication when the communication stop is done. (**Figure 27.50**).

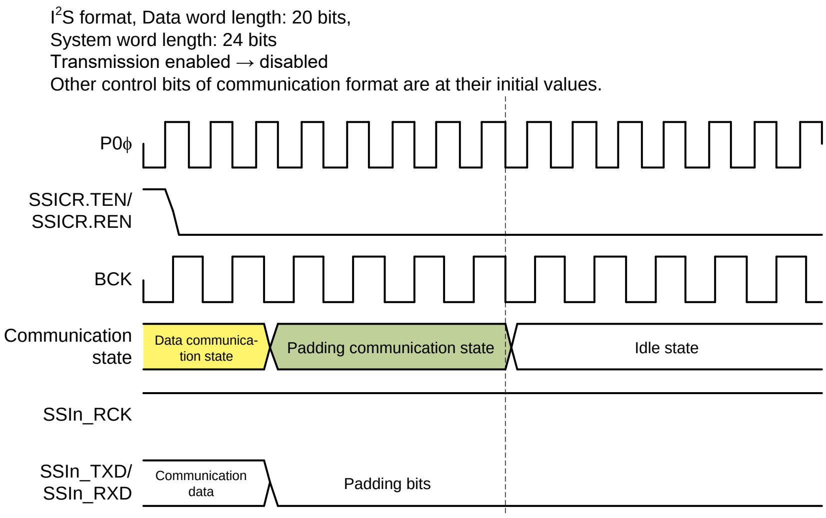


Figure 27.50 Halt from the Padding Communication State

27.5.2 Communication Operation

Figure 27.51 shows the communication flow of SSIF-2.

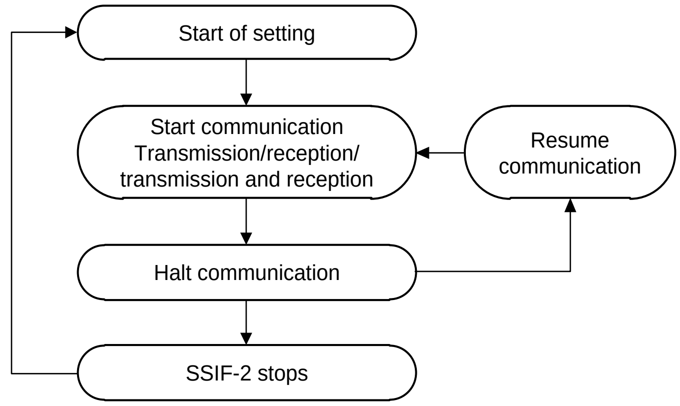
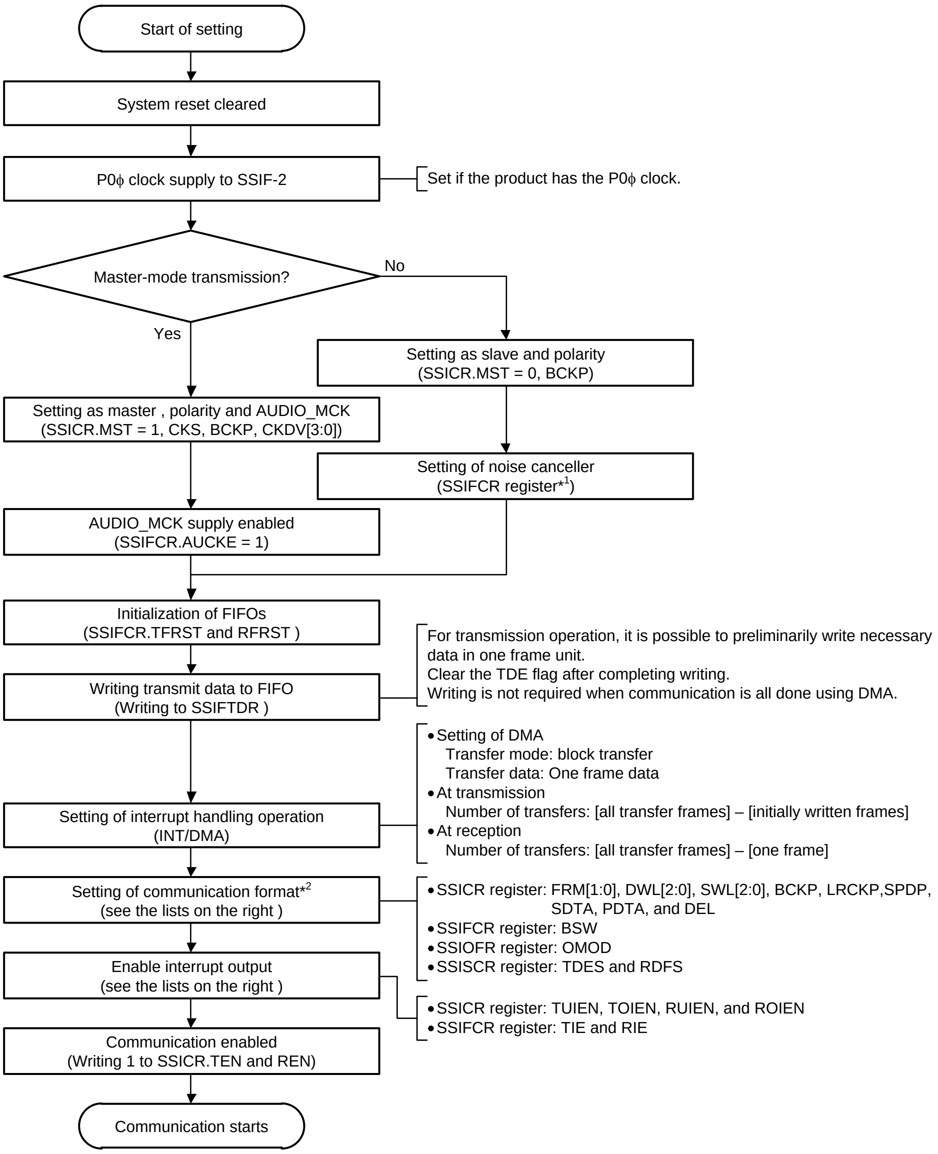


Figure 27.51 SSIF-2 Communication Operation

The procedure of each operation is described in **Section 27.5.2.1, Start Communication** to **Section 27.5.2.7, Resume Communication**.

27.5.2.1 Start Communication

This section describes how to start communication of SSIF-2. **Figure 27.52** shows the procedure to start communication. Be sure to follow the procedure. See **Section 27.5.2.2** for transmission operation and **Section 27.5.2.3** for reception operation.



Note 1. BCKNCE, LRCKNCE, LRCKNCE
Change all the settings of the noise cancellation to the same setting.

Note 2. Set SSICR.MUEN, SSIOFR.BCKASTP, and SSIOFR.LRCONT after completing the setting of the communication format.

Figure 27.52 Procedure to Start Communication (CPU Procedure)

For SSIF-2 communication, continuous communication is possible with DMA interrupt handling. For transmission, write 1 to SSIFCR.TIE, SSICR.TUIEN, and SSICR.TOIEN. For reception, write 1 to SSIFCR.RIE, SSICR.RUIEN, and SSICR.ROIEN.

27.5.2.2 Transmission

Follow the transmission operation procedure (**Figure 27.53**) while the transmission is operating.

Transmission starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while transmission is enabled (SSICR.TEN = 1 and SSICR.REN = 0) and while there is one or more frame of serial data in the transmit FIFO data register (SSIFTDR).

SSIF-2 outputs a transmit data empty interrupt to DMA according to the setting condition of TDE (SSISCR.TDES) and transmit data empty interrupt (SSIFCR.TIE) being enabled.

This interrupt requests writing to the transmit FIFO data register (SSIFTDR). By writing to SSIFTDR at each generation of an interrupt, serial data transmission is realized.

In the procedure to start communication, set DMA handling for transmit data empty interrupts as writing to the transmit FIFO data register (SSIFTDR). As a result, SSIF-2 is able not to mediate CPU and to transmit continuous data. When it becomes empty capacity of transmission FIFO data register set with SSISCR.TDES, transmission data empty interrupt is generated. Set the writing number of times according to empty capacity of transmission FIFO data register that transmission data empty interrupt shows. In a case of an error, stop receive procedure (**Section 27.5.2.5**) and then take the error procedure (**Section 27.5.2.6**).

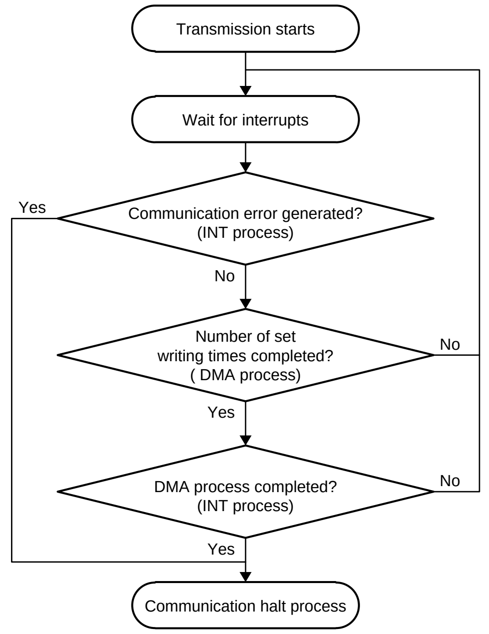


Figure 27.53 Transmission Procedure

NOTE

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe one of SSIFSR.TDE, and write it in SSIFTDR. Follow empty capacity of transmission FIFO data register set with SSISCR.TDES about the number of times written in SSIFTDR detecting 1 of SSIFSR.TDE. After writing the transmission data corresponding to the free space to SSIFTDR, polling ‘1’ of SSIFSR.TDE and repeating this enables continuous transmission operation.

27.5.2.3 Reception

Follow the reception operation procedure (**Figure 27.54**) while the reception is operating.

Reception starts in synchronization with the start trigger of the SSILRCK/SSIFS signal while reception is enabled (SSICR.TEN = 0 and SSICR.REN = 1). SSIF-2 outputs a receive data full interrupt to DMA according to the setting of RDF (SSISCR.RDFS) and receive data empty interrupt (SSIFCR.RIE) being enabled. This interrupt requests reading from the receive FIFO data register (SSIFRDR). By reading from SSIFRDR at each generation of an interrupt, serial data reception is realized.

In the procedure to start communication, set DMA handling for receive data full interrupts as reading from receive FIFO data register (DDIFRDR). As a result, SSIF-2 is able not to mediate CPU and to receive continuous data. If the data capacity of reception FIFO data register set with SSISCR.RDFS is stored, receive data full interrupt is generated. Set the writing number of times according to the data capacity of reception FIFO data register that receive data full interrupt shows.

In the case of an error, stop the receive operation (**Section 27.5.2.6**) and then take the error procedure (**Section 27.5.2.6**).

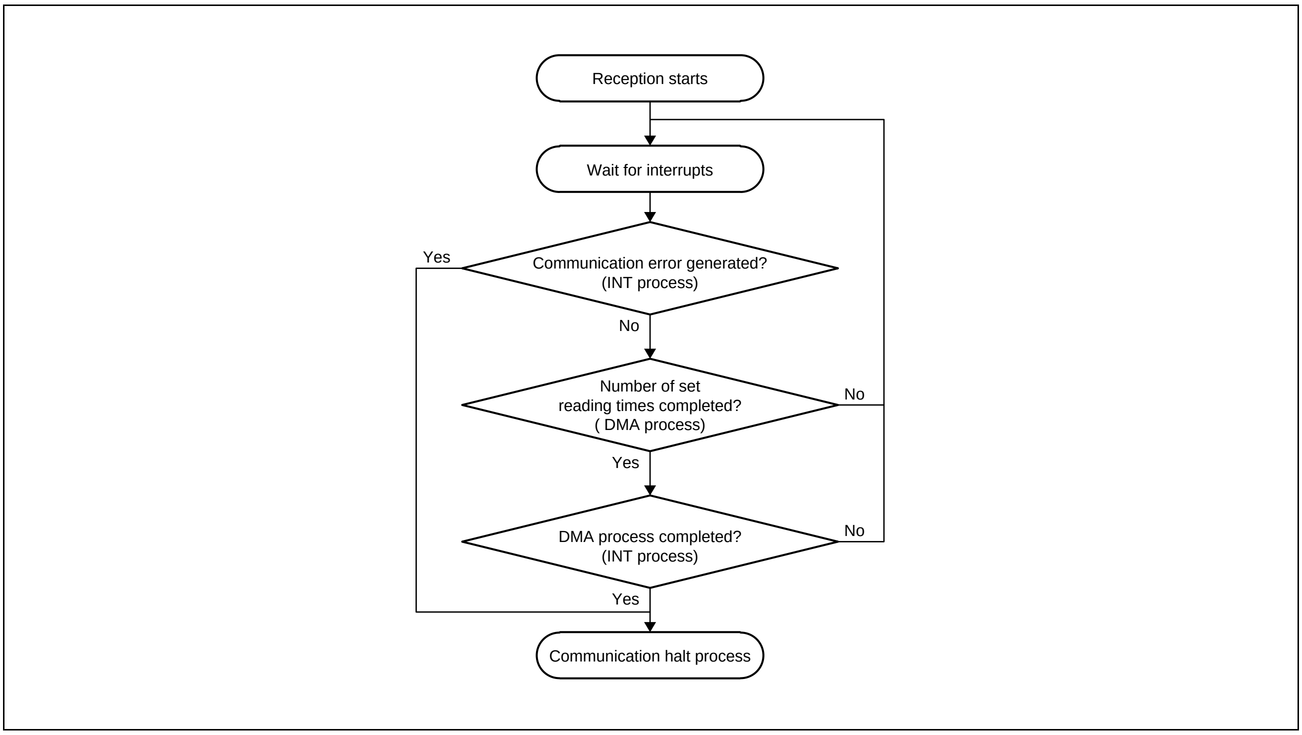


Figure 27.54 Reception Procedure

NOTE

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe 1 of SSIFSR.RDF, and read SSIFRDR. Follow the receive data capacity of reception FIFO data register set with SSISCR.RDFS about the number of times in which SSIFRDR is read detecting 1 of SSIFSR.RDF. After reading the received data from SSIFRDR, Continuous reception operation is possible by polling ‘1’ of SSIFSR.RDF and repeating this.

27.5.2.4 Transmission and Reception

Transmission and reception start in synchronization with the start trigger of the SSILRCK/SSIFS signal while transmission and reception are enabled (SSICR.TEN = 1 and SSICR.REN = 1) and while there is one or more frames of serial data in the transmit FIFO data register (SSIFTDR). The transmission and reception operation of SSIF-2 can be transmitted and received continuously by each doing the procedure for showing in the transmission operation (**Section 27.5.2.2**) and the reception operation (**Section 27.5.2.3**). As transmission and reception are independent in SSIF-2, see sections of transmission and reception respectively. See **Section 27.5.2.5, Halt Communication** to halt communication.

27.5.2.5 Halt Communication

This section describes how to halt communication of SSIF-2. **Figure 27.55** shows the procedure to halt communication. Be sure to follow the procedure.

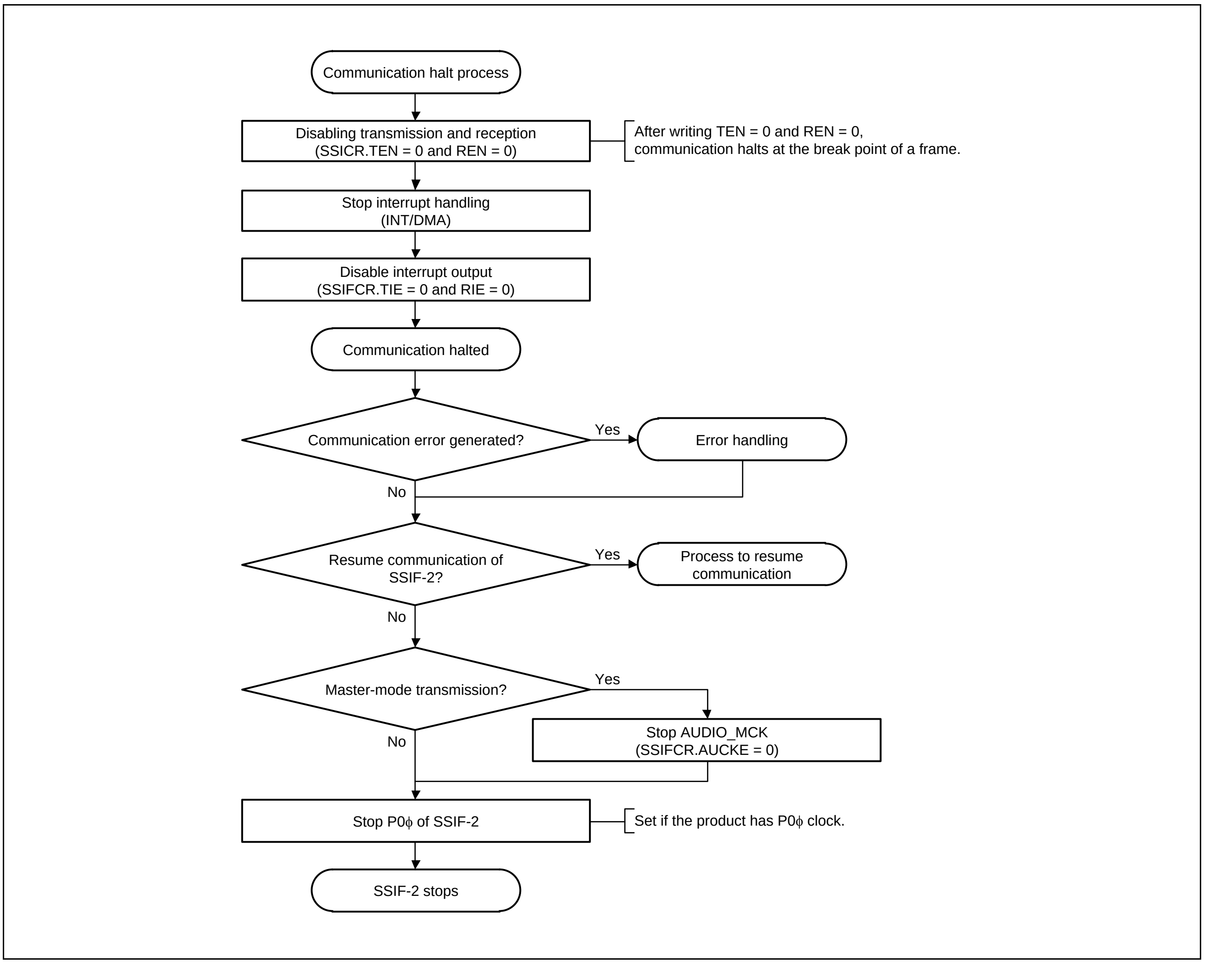


Figure 27.55 Procedure to Halt Communication (CPU Procedure)

To halt the communication of SSIF-2, supply of the following clocks are required until the SSISR.IIRQ bit indicates an idle state.

- Input clock from the SSIBCK pin when SSICR.MST = 0
- AUDIO_MCK when SSICR.MST = 1

To resume communication of SSIF-2 in the previous setting, see **Section 27.5.2.7, Resume Communication**.

NOTE

When communication of SSIF-2 is halted according to the procedure to halt communication in **Figure 27.55**, resume communication according to the procedure to start communication in **Figure 27.52**.

27.5.2.6 Error Handling

SSIF-2 has the following four errors.

- Transmit underflow error
- Transmit overflow error
- Receive underflow error
- Receive overflow error

When an underflow error or overflow error is generated, SSIF-2 need to be restarted. Take the error procedure (**Figure 27.56**) according to the procedure to halt communication (**Figure 27.55**).

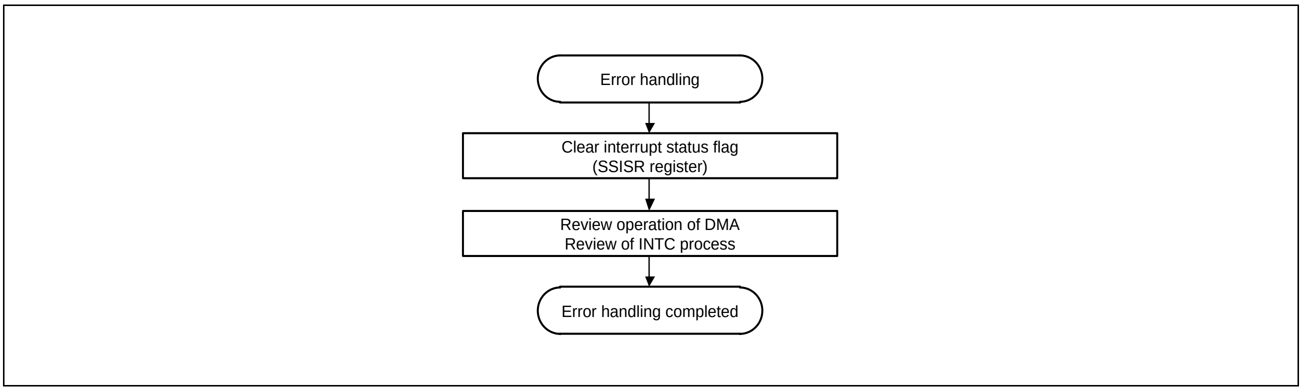


Figure 27.56 Error Handling Procedure

Four error operations are described as follows. When the interrupt output enable bit of the SSICR register is enabled and error flags are set, an error interrupt is generated. See descriptions of flags in **Section 27.4.1.2, Status Register (SSISR)** for the setting conditions of error flags.

(1) Transmit Underflow Error

In the case a transmit underflow error is generated during transmission (SSICR.TEN = 1, SSISR.IIRQ = 0), review the number of write operations to the transmit FIFO data register (SSIFTDR) according to serial data empty interrupts. After generation of a transmit underflow error, data to be transmitted from SSIF-2 are 0 output. To transmit data written to SSIFTxD pin normally, run the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**). When this error is generated, serial data is consumed as usual. To resume transmission of SSIF-2, write data from the beginning of one frame.

(2) Transmit Overflow Error

In the case a transmit overflow error is generated, review the number of write operations to the transmit FIFO data register (SSIFTDR) according to transmit data empty interrupts. Data is not written to SSIFTDR where a transmit overflow error is generated. This error is generated regardless of while the transmission is operating. To resume transmission of SSIF-2, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**). When communication is resumed, consider the lost serial data.

(3) Receive Underflow Error

In the case a receive underflow error is generated, review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. The value read from SSIFRDR is undefined where a receive underflow error is generated. This error is generated regardless of while the reception is operating. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**).

(4) Receive Overflow Error

In the case a receive overflow error is generated during reception (SSICR.REN = 1, SSISR.IIRQ = 0), review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. Data is not stored in SSIFRDR when a receive overflow error is generated. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 27.55**) and then take the error handling procedure (**Figure 27.56**).

27.5.2.7 Resume Communication

To resume communication of SSIF-2, follow the procedure shown in **Figure 27.57**. After communication is halted, the same setting is applied when it is resumed. To change the settings of clocks and slave/master mode, follow the procedure to start communication (**Figure 27.52**). For communication after resume, see **Section 27.5.2.2** and **Section 27.5.2.3**.

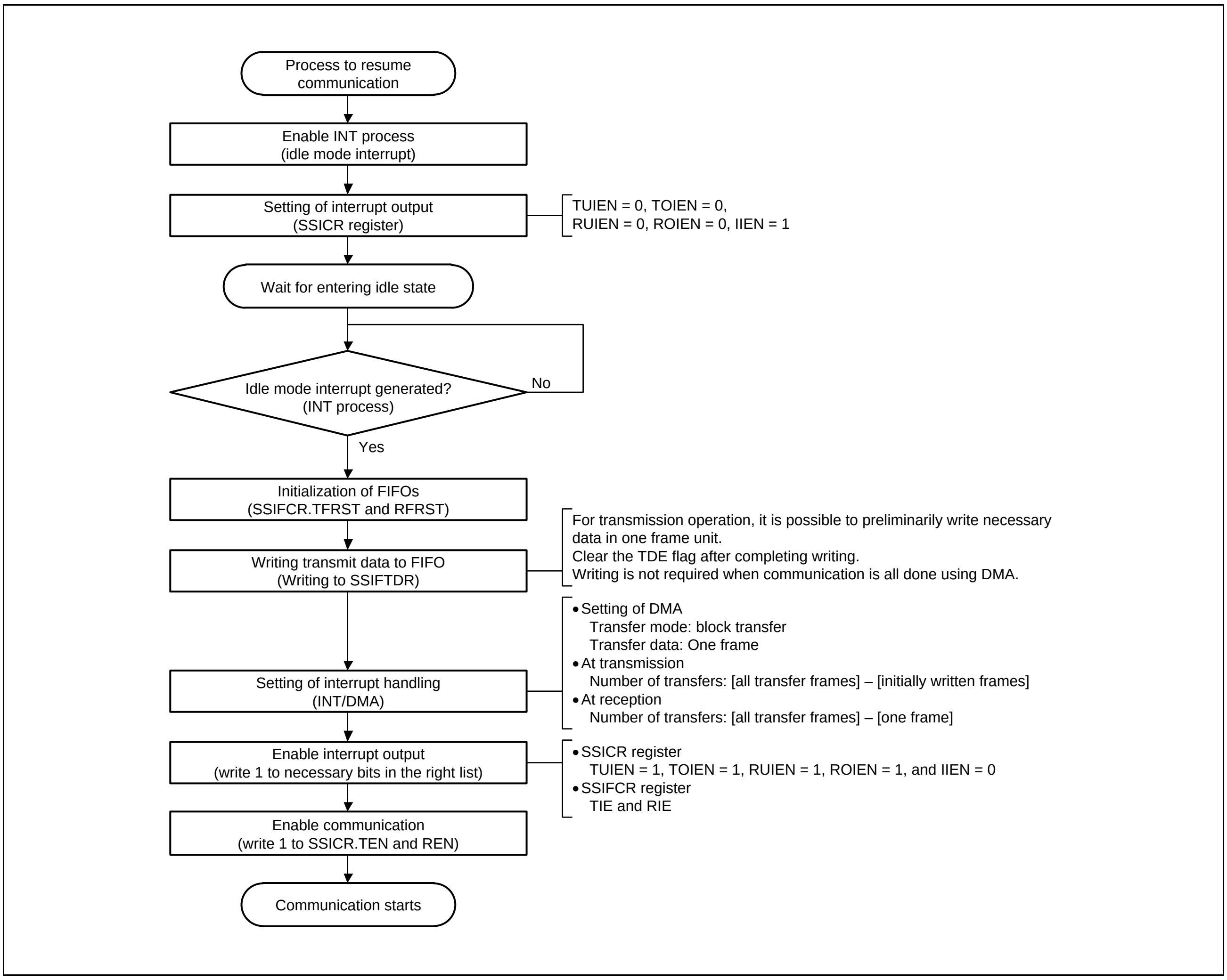


Figure 27.57 Procedure to Resume Communication (CPU Procedure)

27.5.3 Interrupt Sources

Table 27.18 lists the interrupt sources. Set enable/disable of interrupt output of each source with the TUIEN, TOIEN, RUIEN, ROIEN, and IIEN bits in the SSICR register and the TIE and RIE bits in the SSIFCR register.

Table 27.18 Interrupt Sources

Interrupt Source	Interrupt	Interrupt Flag	Interrupt Sense	DMA Activation
INT_ssif_int_req	Transmit underflow interrupt/ Transmit overflow interrupt/ Receive underflow interrupt/ Receive overflow interrupt/ Idle mode interrupt	SSISR.TUIRQ SSISR.TOIRQ SSISR.RUIRQ SSISR.ROIRQ SSISR.IIRQ	Level	Not available
INT_ssif_dma_tx	Transmit data empty interrupt	SSIFSR.TDE	Edge	Available
INT_ssif_dma_rx	Receive data full interrupt	SSIFSR.RDF	Edge	Available
INT_ssif_dma_rt	Receive data full interrupt/ Transmit data empty interrupt	SSIFSR.TDE/SSIFSR.RDF	Edge	Available

27.5.3.1 INT_ssif_int_req Interrupt

This interrupt source combines five interrupts. Enable output of necessary interrupts before using SSIF-2. This interrupt moves from the flag and the output enable permission that five factors have respectively (**Figure 27.58**). To clear an interrupt, set the interrupt enable to 0 or clear the interrupt flag to 0.

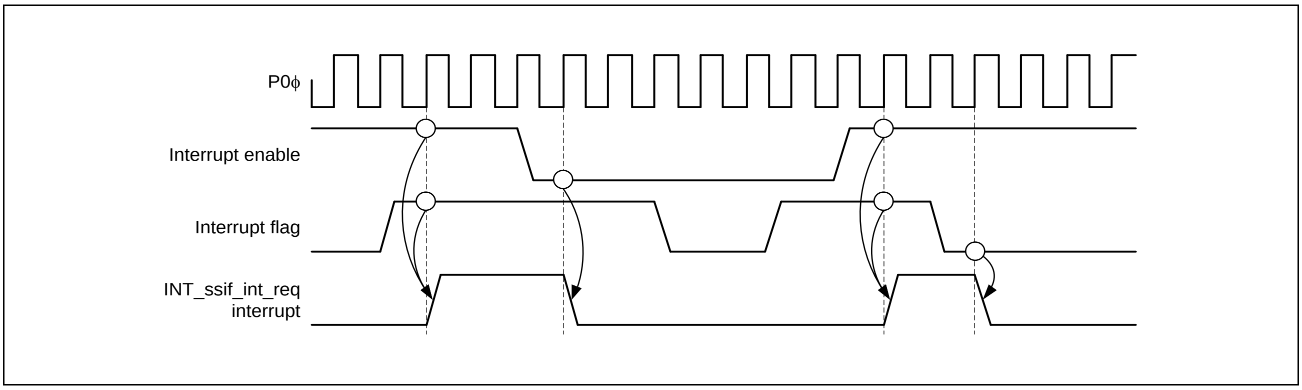


Figure 27.58 Timing Chart of the Common Interrupt Source, INT_ssif_int_req

(1) Transmit underflow interrupt

As the transmit underflow interrupt, SSISR.TUIRQ is output while SSICR.TUIEN = 1. To restart transmission (SSICR.TEN = 1), enable the output of this interrupt (SSICR.TUIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**).

(2) Transmit overflow interrupt

As the transmit overflow interrupt, SSISR.TOIRQ is output while SSICR.TOIRQ = 1. To restart transmission (SSICR.TEN = 1), enable the output of this interrupt (SSICR.TOIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**).

(3) Receive underflow interrupt

As the receive underflow interrupt, SSISR.RUIRQ is output while SSICR.RUIRQ = 1. To restart reception (SSICR.REN = 1), enable the output of this interrupt (SSICR. RUIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**).

(4) Receive overflow interrupt

As the receive overflow interrupt, SSISR.ROIRQ is output while SSICR.ROIRQ = 1. To restart reception (SSICR.REN = 1), enable the output of this interrupt (SSICR. ROIRQ = 1). In the case this interrupt is generated, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**).

(5) Idle mode interrupt

As the idle mode interrupt, SSISR.IIRQ is output while SSICR.IIEN = 1. When the thing that the communication has stopped completely is confirmed, this interrupt is used (**Figure 27.57**).

27.5.3.2 INT_ssif_dma_tx Interrupt [full-duplex communication]

As the transmit data empty interrupt, a pulse interrupt is output by the following condition.

- SSIFSR.TDE and SSIFCR.TIE becomes 1.
  SSIF-2 operation: SSIFSR.TDE becomes 1 from 0 while SSIFCR.TIE = 1
  CPU operation: SSIFCR.TIE is changed to 1 from 0 while SSIFSR.TDE = 1

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable (**Figure 27.59**).

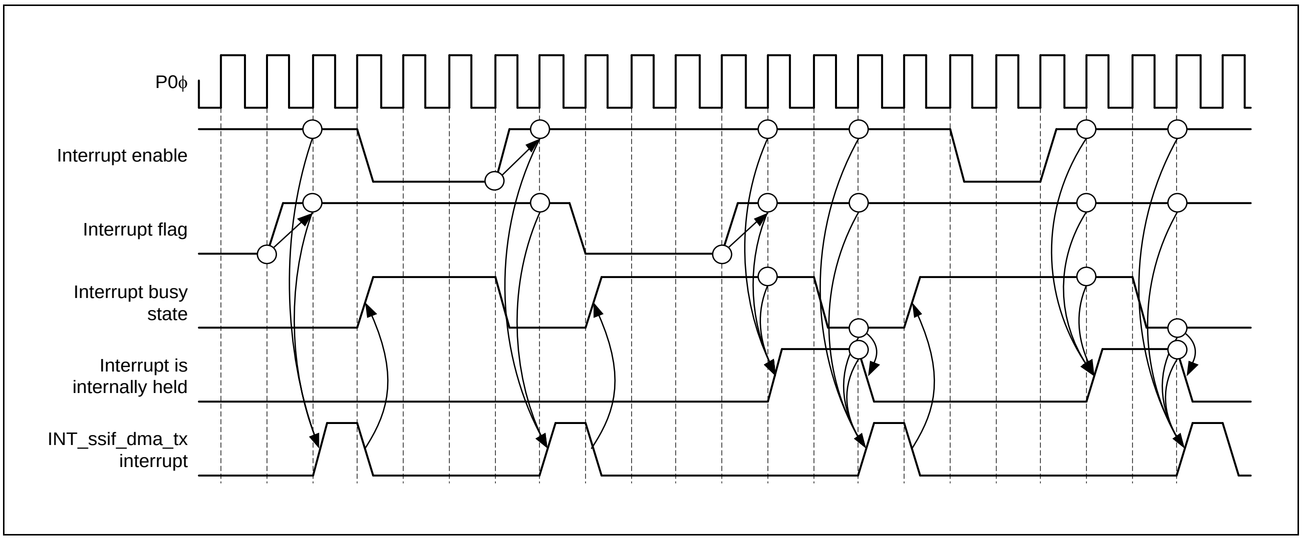


Figure 27.59 INT_ssif_dma_tx Interrupt Timing Chart

27.5.3.3 INT_ssif_dma_rx Interrupt [full-duplex communication]

As the receive data full interrupt, a pulse interrupt is output by the following condition.

- SSIFSR.RDF and SSIFCR.RIE becomes 1.
  SSIF-2 operation: SSIFSR.RDF becomes 1 from 0 while SSIFCR.RIE = 1.
  CPU operation: SSIFCR.RIE is changed to 1 from 0 while SSIFSR.RDF = 1.

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable. It operates in the same way as in **Figure 27.59**.

27.5.3.4 INT_ssif_dma_rt Interrupt [half-duplex communication]

This interrupt is output by two sources, transmit data empty interrupt and receive data full interrupt. When this interrupt is generated, read the interrupt flag and specify the interrupt source.

An interrupt suppression function is available for this interrupt to prevent the thing that cannot be taken. When the DMA is busy state (Interrupt cannot be accepted), and the interruption condition occurs, output of this interrupt is held. This interrupt is output after the DMA becomes acceptable (**Figure 27.59**).

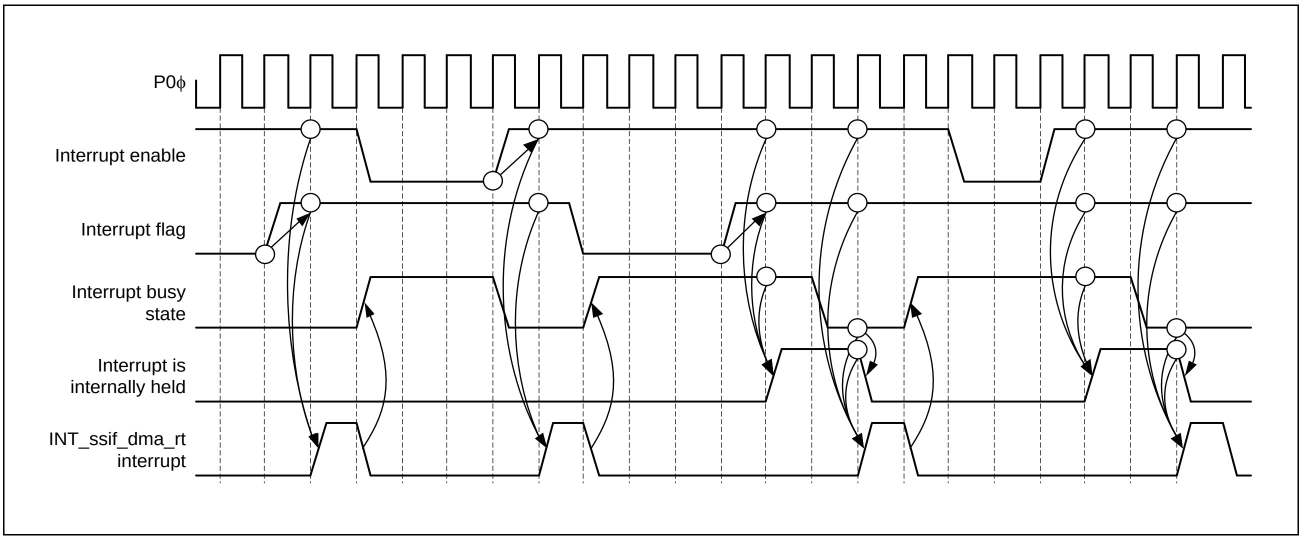


Figure 27.60 INT_ssif_dma_rt Interrupt Timing Chart

27.5.4 Software Resets

SSIF-2 has three software reset bits to reset its states.

- SSIF-2 software reset (SSIFCR.SSIRST)
- Transmit FIFO data register reset (SSICR.TFRST)
- Receive FIFO data register reset (SSIFCR.RFRST)

It explains the procedure of three software reset.

27.5.4.1 Software Reset Procedure

(1) SSIF-2 Software Reset

For the SSIF-2 software reset bit (SSIFCR.SSIRST), follow the procedure shown in **Figure 27.61**. After a reset, the same setting is applied when it is resumed. To change the settings of clocks and slave/master mode, follow the procedure to start communication (**Figure 27.52**). See **Section 27.5.2.2, Transmission** and **Section 27.5.2.3, Reception** respectively for transmission and reception after communication is resumed.

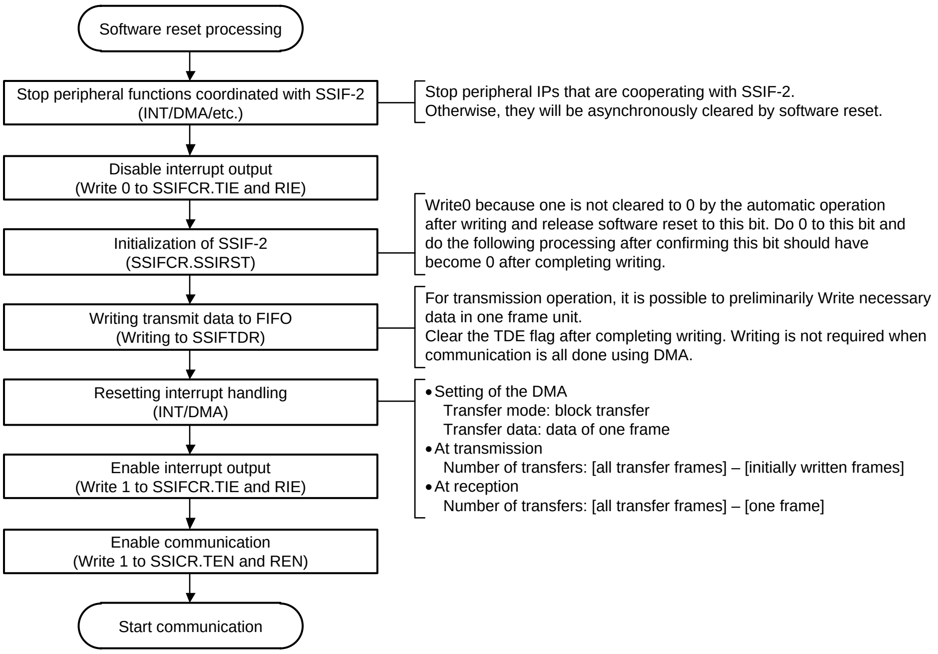


Figure 27.61 Software Reset Procedure (CPU Procedure)

(2) Transmit FIFO Data Register Reset

To initiate a transmit FIFO data register reset, follow the procedure to start communication (**Figure 27.52**) and resume communication (**Figure 27.57**).

(3) Receive FIFO Data Register Reset

To initiate a receive FIFO data register reset, follow the procedure to start communication (**Figure 27.52**) and resume communication (**Figure 27.57**).

27.6 Notes

27.6.1 Notes

27.6.1.1 Attention by communication using DMA

SSIF-2 assumes the DMA access as follows.

Table 27.19 Interrupt to expect DMA access

Communication operation	Interrupt	DMA access	LSI specification	Remarks
Transmission	INT_ssif_dma_tx	Write to SSIFTDR	Full duplex transmission	
	INT_ssif_dma_rt		Half duplex transmission	
Reception	INT_ssif_dma_rx	Read SSIFRDR	Full duplex transmission	
	INT_ssif_dma_rt		Half duplex transmission	
Transmission and Reception	INT_ssif_dma_tx	Write to SSIFTDR	Full duplex transmission	
	INT_ssif_dma_rx	Read SSIFRDR	Half duplex transmission	

27.6.1.2 Notes for Slave-mode Communication

(1) ADCKE Control

In slave-mode communication (SSICR.MST = 0), SSIF-2 needs supply of SSIBCK. To stop BCK on the master side, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1). If BCK is stopped before SSIF-2 becomes idle, take the procedure to start communication (**Figure 27.52**) or wait for an idle state by taking the procedure to resume communication (**Figure 27.57**).

(2) SSI_RCK

SSIF-2 has the terminal SSI_RCK that shows the synchronization of the communication. Match communication format of the opposing device and SSIF-2 when SSIF-2 is slave communication (SSICR.MST=0). SSIF-2 uses the signal of SSI_RCK only as a communication start trigger.

27.6.1.3 Notes for Master-mode Communication

(1) ADCKE Control

In master-mode communication (SSICR.MST = 1), SSIF-2 operates with the audio clock (AUDIO_MCK). To stop SSIF- 2 completely, make sure that SSIF-2 is in the idle state (SSISR.IIRQ = 1) and then write 0 to SSIFCR.ADCKE. If 0 is written to SSIFCR.ADCKE before SSIF-2 becomes idle, take the procedure to start communication (**Figure 27.52**).

(2) LRCONT Control

To stop the output to the SSI_RCK pin with SSIOFR.LRCONT when SSIF-2 is in the idle state in master-mode communication (SSICR.MST = 1), note the following: The output stops when 0 is written from 1 to SSIOFR.LRCONT. (**Figure 27.43**). Make sure that the remote device is not affected.

(3) BCKASTP Control

To stop the output to the SSIBCK pin with SSIOFR.BCKASTP in master-mode communication (SSICR.MST = 1) and while SSIF-2 is in the idle state, note the following; By writing 0 to SSIOFR.BCKASTP while it is 1, the output stops immediately (**Figure 27.44**). So, make sure that the remote device is not affected.

NOTE

When the opposing device who is the slave needs the clock of the terminal SSI_BCK before the communication operates, it is not possible to use it.

27.6.1.4 Notes for Communication Flow

(1) When an Error Interrupt is Generated

SSIF-2 has the following four errors.

- Transmit underflow error
- Transmit overflow error
- Receive underflow error
- Receive overflow error

When an underflow error or overflow error is generated, SSIF-2 need to be restarted. Take the error procedure (**Figure 27.56**) according to the procedure to halt communication (**Figure 27.55**).

(a) Transmit Underflow Error

In the case a transmit underflow error is generated during transmission (SSICR.TEN = 1, SSISR.IIRQ = 0), review the number of write operations to the transmit FIFO data register (SSIFTDR) according to serial data empty interrupts. After generation of a transmit underflow error, data to be transmitted from SSIF-2 are 0 output. To transmit data written to SSIFTxD pin normally, run the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**). When this error is generated, serial data is consumed as usual. To resume transmission of SSIF-2, write data from the beginning of one frame.

(b) Transmit Overflow Error

In the case a transmit overflow error is generated, review the number of write operations to the transmit FIFO data register (SSIFTDR) according to transmit data empty interrupts. Data is not written to SSIFTDR where a transmit overflow error is generated. This error is generated regardless of while the transmission is operating. To resume transmission of SSIF-2, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**). When communication is resumed, consider the lost serial data.

(c) Receive Underflow Error

In the case a receive underflow error is generated, review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. The value read from SSIFRDR is undefined where a receive underflow error is generated. This error is generated regardless of while the reception is operating. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 27.55**) and then take the error procedure (**Figure 27.56**).

(d) Receive Overflow Error

In the case a receive overflow error is generated during reception (SSICR.REN = 1, SSISR.IIRQ = 0), review the number of reading operations from the receive FIFO data register (SSIFRDR) according to receive data full interrupts. Data is not stored in SSIFRDR when a receive overflow error is generated. To resume reception of SSIF-2, take the procedure to halt communication (**Figure 27.55**) and then take the error handling procedure (**Figure 27.56**).

(2) Transmit Data Empty Interrupt

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe one of SSIFSR.TDE, and write it in SSIFTDR. Follow empty capacity of transmission FIFO data register set with SSISCR.TDES about the number of times written in SSIFTDR detecting 1 of SSIFSR.TDE. Clear the transmission data that corresponds to empty capacity and clear the SSIFSR.TDE flag to SSIFTDR after writing. Transmission operation consecutive by the repeated thing is possible. SSIFSR.TDE is not cleared when not clearing.

(3) Receive Data Full Interrupt

The communication flow defined in SSIF-2 uses the DMA. In the case the DMA is not used, observe 1 of SSIFSR.RDF, and read SSIFRDR. Follow the receive data capacity of reception FIFO data register set with SSISCR.RDFS about the number of times in which SSIFRDR is read detecting 1 of SSIFSR.RDF. Clear data that does the reception completion and clear the SSIFSR.RDF flag from SSIFRDR after reading. Reception operation consecutive by the repeated thing is possible. SSIFSR.RDF is not cleared when not clearing.

(4) Switching Transfer Modes

1. For state transition from transmission, reception, and transmission and reception, disable transmission and reception (SSICR.TEN = 0, SSICR.REN = 0).
2. Confirm it is in the idle state (SSISR.IIRQ = 1).
3. In the idle state, set the SSICR.TEN bit and the SSICR.REN bit again and resume transfer.

(5) Resume Communication after Halting SSIF-2

When communication of SSIF-2 is halted according to the procedure to halt communication in **Figure 27.55**, resume communication according to the procedure to resume communication in **Figure 27.52**.

27.6.1.5 Write Access Restriction

(1) SSICR Register

If the TEN bit or REN bit is rewritten, make sure that the SSISR.IIRQ bit is in the desired status. If written, the operation performed immediately after writing is not guaranteed. For example, after enabling operation, make sure that SSISR.IIRQ = 0 and after disabling operation, make sure that SSISR.IIRQ = 1.

(a) TEN Bit and REN Bit

Writing 1 to these bits, transmission starts in synchronization with start trigger. Refer to **Section 27.5.2.2, Transmission**, **Section 27.5.2.3, Reception**, and **Section 27.5.2.4, Transmission and Reception** for details. Writing 0 to these bits, transmission stops in synchronization with next frame boundary. Set 1 at the same time when you use SSIF-2 as transmission and reception. To stop communication, make sure to stop transmission and reception (TEN = 0 and REN = 0).

(2) SSICR Register

(a) Clear operation of TUIRQ and TOIRQ

After the communication is permitted (SSICR.TEN is written and ‘1’ is written from ‘0’), the transmission error flag (TOIRQ and TUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the transmission error flag might not be able to be read.

(b) Clear operation of RUIRQ and ROIRQ

After the communication is permitted (SSICR.REN is written and ‘1’ is written from ‘0’), the transmission error flag (ROIRQ and RUIRQ of the SSISR register) is cleared. However, when the SSISR register is continuously read, the clearness of the reception error flag might not be able to be read.

(3) Communication State

Writing to the bits with shaded area in **Table 27.20** is prohibited. If written, the operation performed immediately after writing is not guaranteed.

Table 27.20 Bits Protected from Writing during Communication

Symbol	Address (Base+)		+0								+1							
			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSICR	H'00	+0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	FRM[1:0]		DWL[2:0]			SWL[2:0]		
		+2	—	MST	BCKP	LRCKP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]				MUEN	—	TEN	REN
SSISR	H'04	+0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SSIFCR	H'10	+0	AUCKE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSIRST
		+2	—	—	—	—	BSW	BCKNCE	LRCKNCE	RxDNCE	—	—	—	—	TIE	RIE	TFRST	RFRST
SSIFSR	H'14	+0	—	—	TDC[5:0]						—	—	—	—	—	—	—	TDE
		+2	—	—	RDC[5:0]						—	—	—	—	—	—	—	RDF
SSIFTDR	H'18	+0	FTDR[31:16]															
		+2	FTDR[15:0]															
SSIFRDR	H'1C	+0	FRDR[31:16]															
		+2	FRDR[15:0]															
SSIOFR		+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	—	—	—	—	BCKASTP	LRCONT	—	—	—	—	—	—	OMOD[1:0]	
SSISCR	H'24	+0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
		+2	—	—	TDES[5:0]						—	—	RDFS[5:0]					

28. CANFD Interface (RS-CANFD)

This section contains a generic description of the CANFD interface (RS-CANFD).

The first part of this section describes all this LSI specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of RS-CANFD.

28.1 Features of RS-CANFD

28.1.1 Number of Units and Channels

This product has the following number of RS-CANFD units.

Table 28.1 Number of Units

RS-CANFD	
Number of Units	1
Name	RSCFD0

This product has the CANFD interface channel listed below.

Table 28.2 Unit Configurations and Channels

RS-CANFD	
Number of Channels	2
Name	CAN0, CAN1

The RS-CANFD has two interface modes (classical CAN mode and CANFD mode) and uses different registers for each mode. There are two types of register names RSCANnXXX and RSCFDnCFDXXX (XXX: arbitrary) depending on interface modes. When explaining specifications common to two registers, register names are described as RSCFDn(CFD)XXX.

Table 28.3 Index

Index	Meaning
n	Throughout this section, the individual RS-CANFD units are generically indicated by the index “n” (n = 0); for example, RSCFDn(CFD)GCTR is the global control register of the RSCFDn unit.
m	Throughout this section, the individual channels of RS-CANFD units are generically indicated by the index “m” (m = 0, 1); for example, RSCFDn(CFD)CmSTS is the channel m status register.
j	The individual registers associated with receive rule table are generically indicated by the index “j” (j = 0 to 15); for example, RSCFDn(CFD)GAFLIDj is the receive rule ID register.
k	The individual transmit/receive FIFO buffers are generically indicated by the index “k” (k = 0 to 5); for example, RSCFDn(CFD)CFCCk is the transmit/receive FIFO buffer configuration/control register.
x	The individual receive FIFO buffers are generically identified by the index “x” (x = 0 to 7); for example, RSCFDn(CFD)RFSTSx is the receive FIFO buffer status register.
d	Data field registers of transmit/receive FIFO buffers and receive FIFO buffers are identified by “d” (classical CAN mode: d = 0, 1, CANFD mode: d = 0 to 15). For example, the transmit/receive FIFO buffer data field register is described as RSCFDn(CFD)CFDFd_k.
q	The individual receive buffers are generically indicated by the index “q” (q = 0 to 31); for example, RSCFDn(CFD)RMIDq is the receive buffer ID register.
p	The individual transmit buffers are generically indicated by the index “p” (p = 0 to 31); for example, RSCFDn(CFD)TMCp is the transmit buffer control register.
b	Data field registers of receive buffers and transmit buffers are identified by “b” (classical CAN mode: b = 0, 1, CANFD mode: b = 0 to 4). For example, the receive buffer data field register is described as RSCFDn(CFD)RMDFb_q.
r	The individual RAM tests for CAN are generically indicated by the index “r” (r = 0 to 63); for example, RSCFDn(CFD)RPGACCr is the RAM test page access register.
y	The registers not covered above are indicated by the letter “y” (y = 0); for example, RSCFDn(CFD)RMNDy is a receive buffer new data register.

Note: The functions and descriptions of registers in this section are for the RS-CANFDs that has 2 channels (m = 0, 1). When referring to information with indexes, regard the index values as the ones corresponding to your target product. Also, note that, if the value of an index exceeds the range described in this section due to your target product, write the value after reset when writing to bits outside the index range.

28.1.2 Register Base Address

RSCFDn base addresses are listed in the following table.

RSCFDn register addresses are given as offsets from the base addresses in general.

Table 28.4 Register Base Address

Base Address Name	Base Address
<RSCFD0_base>	H'0_1005_0000 (Cortex-A55 Address Space) H'4005_0000 (Cortex-M33 Address Space Non-Secure) H'5005_0000 (Cortex-M33 Address Space Secure)

28.1.3 Clock Supply

The RSCFDn clock supply is shown in the following table.

Table 28.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
RSCFDn	clk_xincan	CAN_CLK
	clkc	P0ϕ/2
	pclk	P0ϕ

The operating frequency of the RSCFDn depends on the transfer rate and the number of channels in use. **Table 28.6** shows the range of the frequency.

Table 28.6 Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI

Condition			Range of Operating Frequency		
Mode	Transfer Rate	No. of Channels in Use	pclk	clk_xincan*[1]	clkc*[1] *[2]
CANFD	4 Mbps	2ch	pclk ≥ 64 MHz	32 MHz	32MHz
		1ch			
	2 Mbps	2ch	pclk ≥ 32 MHz	16 MHz ≤ clk_xincan ≤ pclk/2	16 MHz ≤ clkc ≤ pclk/2
		1ch			
CAN/CANFD	1 Mbps	2ch	pclk ≥ 30.6 MHz	8 MHz ≤ clk_xincan ≤ pclk/2	8 MHz ≤ clkc ≤ pclk/2
		1ch	pclk ≥ 27.5 MHz		
	500 kbps	2ch	pclk ≥ 27.5 MHz	4 MHz ≤ clk_xincan ≤ pclk/2	4 MHz ≤ clkc ≤ pclk/2
		1ch			
	125 kbps	2ch	pclk ≥ 27.5 MHz	1 MHz ≤ clk_xincan ≤ pclk/2	1 MHz ≤ clkc ≤ pclk/2
		1ch			

Note 1. Setting the DCS bit in the RSCFDn(CFD)GCFG register enables to select either clk_xincan or clkc. Set clocks less than or equal to pclk/2. CAN clock source uses external clock.

Note 2. When pclk< 25 MHz, select clk_xincan.

28.1.4 Interrupt Request

RSCFDn interrupt requests are listed in the following table.

Table 28.7 Interrupt Requests

Unit Interrupt Signal		Outline
INTRCANGERR		CAN global error interrupt
INTRCANGRECC		CAN receive FIFO interrupt
CAN0	INTRCAN0ERR	CAN0 error interrupt
	INTRCAN0REC	CAN0 transmit/receive FIFO receive completion interrupt
	INTRCAN0TRX	CAN0 transmit interrupt
CAN1	INTRCAN1ERR	CAN1 error interrupt
	INTRCAN1REC	CAN1 transmit/receive FIFO receive completion interrupt
	INTRCAN1TRX	CAN1 transmit interrupt

28.1.5 External Input/Output Signals

External input/output signals of RSCFDn are listed below.

Table 28.8 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
CANm_RX (m = 0, 1)	CANm receive data input	CANm_RX (m = 0, 1)
CANm_TX (m = 0, 1)	CANm transmit data output	CANm_TX (m = 0, 1)
CANm_RX_DATARATE_EN (m = 0, 1)	CANm receive data phase output	CANm_RX_DATARATE_EN (m = 0, 1)
CANm_TX_DATARATE_EN (m = 0, 1)	CANm transmit data phase output	CANm_TX_DATARATE_EN (m = 0, 1)
CAN_CLK	Clock source for CAN communication	CAN_CLK

28.2 Overview

28.2.1 Functional Overview

Table 28.9 shows the RS-CANFD module specifications. **Figure 28.1** shows the RS-CANFD module block diagram.

Table 28.9 RS-CANFD Module Specifications (1/3)

Item	Specification
Number of channels	2
Protocol	ISO11898-1 compliant Using CANFD frames is selectable by switching interface modes.
Communication speed	Classical CAN mode: ● Maximum 1 Mbps Communication speed (CANm bit time clock) = $\frac{1}{\text{CANm bit time}}$ CANm bit time = CANmTq × Tq count per bit CANmTq = $\frac{\text{(BRP[9:0] bits in the RSCANnCmCFG register + 1)}}{\text{fCAN}}$ CANm bit time = CANmTq × Tq count per bit fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCANnGCFG register) CANFD mode: ● Nominal bit rate: max.1 Mbps, data bit rate: max. 4 Mbps Transmission rate (CANm nominal bit time clock) = $\frac{1}{\text{CANm nominal bit time}}$ Transmission rate (CANm data bit time clock) = $\frac{1}{\text{CANm data bit time}}$ CANm nominal bit time = CANmTq(N) × Tq count per nominal bit CANm data bit time = CANmTq(D) × Tq count per data bit CANmTq(N) = $\frac{\text{(NBRP[9:0] bits in the RSCFDnCFDCmNCFG register + 1)}}{\text{fCAN}}$ CANmTq(D) = $\frac{\text{(DBRP[7:0] bits in the RSCFDnCFDCmDCFG register + 1)}}{\text{fCAN}}$ fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCFDnCFDGCFG register) m = 0, 1 Tq: Time quantum
Buffer	160 buffers in total ● Individual buffers: 32 buffers (16 buffers × 2 channels) Transmit buffer: 16 buffers per channel Transmit queue: Single queue per channel (shared with the transmit buffer; up to 16 buffers allocatable) ● Shared buffers: 128 buffers for all channels Receive buffer: 0 to 32 buffers Receive FIFO buffer: 8 FIFO buffers (up to 128 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 128 buffers allocatable to each)
Reception function	● Receives data frames and remote frames. ● Selects ID format (standard ID, extended ID, or both IDs) to be received. ● Sets interrupt enable/disable for each FIFO. ● Mirror function (reception of messages transmitted from the own CAN node) ● Timestamp function (to record message reception time as a 16-bit timer value)

Table 28.9 RS-CANFD Module Specifications (2/3)

Item	Specification
Reception filter function	● Selects receive messages according to 128 receive rules. ● Sets the number of receive rules (0 to 128) for each channel. ● Acceptance filter processing: Sets ID and mask for each receive rule. ● DLC filter processing: Enables DLC filter check for each acceptance rule.
Receive message transfer function	● Routing function Transfers receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer ● Label addition function Stores label information together with a message in a receive buffer and FIFO buffer.
Transmission function	● Transmits data frames and remote frames. ● Selects ID format (standard ID, extended ID, or both IDs) to be transmitted. ● Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer. ● Selects ID priority transmission or transmit buffer number priority transmission. ● Transmit request can be aborted (possible to confirm with a flag) ● One-shot transmission function
Interval transmission function	Transmit messages at configurable intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)
Transmit queue function	Transmits all stored messages according to the ID priority.
Transmit history function	Stores the history information of transmission-completed messages Adds timestamp (recording message transmission time as a 16-bit timer value) to the history information.
Gateway function	Transmits a received message automatically.
Bus off recovery mode selection	Selects the method for returning from bus off state. ● ISO11898-1 compliant ● Automatic entry to channel halt mode at bus-off entry ● Automatic entry to channel halt mode at bus-off end ● Transition to channel halt mode by program request ● Transition to the error-active state by program request (forcible return from the bus off state)
Error status monitoring	● Monitors CAN protocol errors (stuff error, form error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock). ● Detects error status transitions (error warning, error passive, bus off entry, and bus off recovery) ● Reads the error counter. ● Monitors DLC errors.
Interrupt source	8 sources ● Global Interrupts (2 sources) Receive FIFO interrupt Global error interrupt ● Channel interrupts (3 sources/channel) CANm transmit interrupt (m = 0, 1) – CANm transmit complete interrupt – CANm transmit abort interrupt – CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode) – CANm transmit history interrupt – CANm transmit queue interrupt CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode) CANm error interrupt
CAN stop mode	Reduces power consumption by stopping clock supply to the RS-CANFD module.
CAN clock source	Selects the clkc or the clk_xincan. As for the range of operating frequency, see **Table 28.6**.

Table 28.9 RS-CANFD Module Specifications (3/3)

Item	Specification
Test function	Test function for user evaluation ● Listen-only mode ● Self-test mode 0 (external loopback) ● Self-test mode 1 (internal loopback) ● Restricted operation mode ● RAM test (read/write test) ● Inter-channel communication test [CRC error test enabled]

28.2.2 Interface Modes

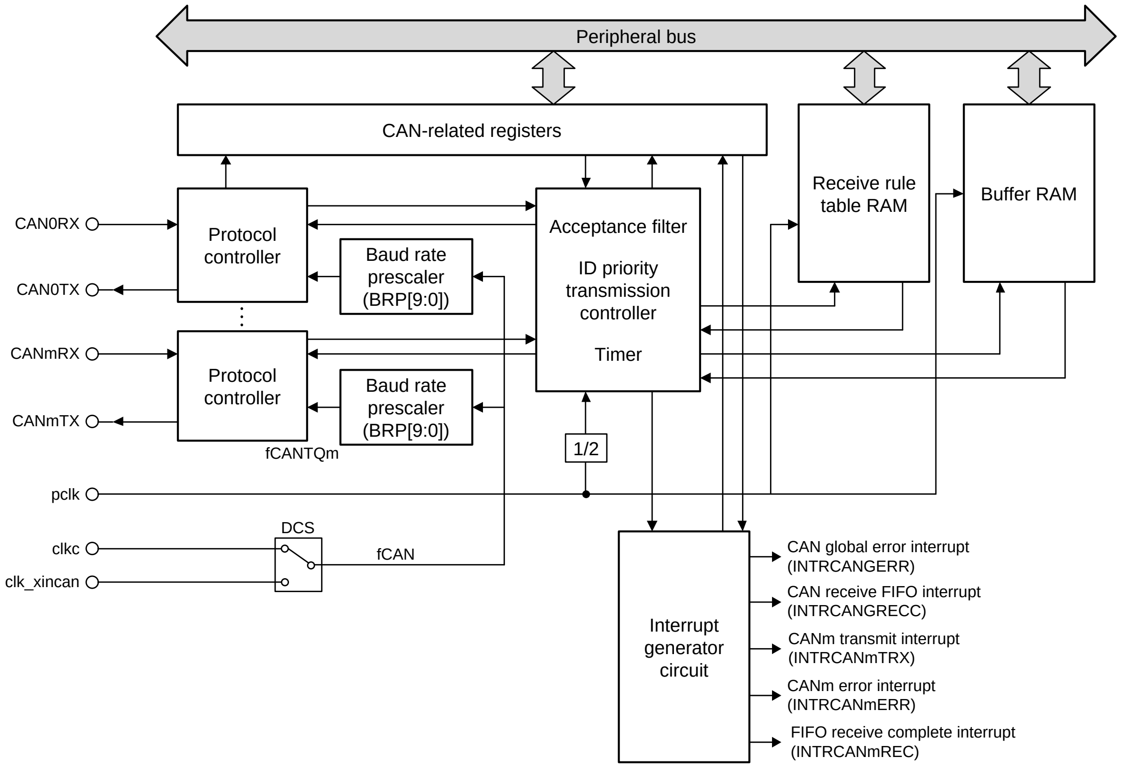
The RS-CANFD has two interface modes.

- Classical CAN mode: Handles only classical CAN frames.
- CANFD mode: Handles classical CAN frames and CANFD frames.

These two modes use different register maps with the same base address. Register maps change by switching interface modes.

Interface modes are switched by the RCMC bit in the RSCFDn(CFD)GRMCFG register.

28.2.3 Block Diagram



Note: m = 0, 1
BRP[9:0]: Bits in the RSCANnCmCFG register
DCS: Bits in the RSCANnGCFG register
fCANTQm: CANmTq clock
fCAN: CAN clock

Figure 28.1 RS-CANFD Module Block Diagram (Classical CAN Mode)

In CANFD mode, different clock signals are input to the baud rate prescaler and the protocol controller respectively. See **Section 28.11.1.3, Communication Speed Setting**.

28.3 Registers (Classical CAN Mode)

28.3.1 List of Registers

The following tables list RS-CANFD registers to be used in classical CAN mode. For details about <RSCFDn_base>, see **Section 28.1.2, Register Base Address**.

For registers initialized in global reset mode and channel reset mode, refer to the following.

- **Table 28.37, Registers Initialized in Global Reset Mode or Channel Reset Mode**
- **Table 28.38, Registers Initialized Only in Global Reset Mode**

Table 28.10 Registers (1/3)

Module	Register	Symbol	Address
Interface mode-related registers			
RSCANn	Global interface mode select register	RSCANnGRMCFG	<RSCFDn_base> + H’04FC
Channel-related registers			
RSCANn	Channel m Configuration Register	RSCANnCmCFG	<RSCFDn_base> + H’0000 + (H’10 × m)
RSCANn	Channel m control register	RSCANnCmCTR	<RSCFDn_base> + H’0004 + (H’10 × m)
RSCANn	Channel m status register	RSCANnCmSTS	<RSCFDn_base> + H’0008 + (H’10 × m)
RSCANn	Channel m error flag register	RSCANnCmERFL	<RSCFDn_base> + H’000C + (H’10 × m)
Global-related registers			
RSCANn	Global configuration register	RSCANnGCFG	<RSCFDn_base> + H’0084
RSCANn	Global control register	RSCANnGCTR	<RSCFDn_base> + H’0088
RSCANn	Global status register	RSCANnGSTS	<RSCFDn_base> + H’008C
RSCANn	Global error flag register	RSCANnGERFL	<RSCFDn_base> + H’0090
RSCANn	Global timestamp counter register	RSCANnGTSC	<RSCFDn_base> + H’0094
RSCANn	Global TX Interrupt Status Register 0	RSCANnGTINTSTS0	<RSCFDn_base> + H’0460
RSCANn	Global FD Configuration Register	RSCANnGFDCFG	<RSCFDn_base> + H’0474
Receive rule-related registers			
RSCANn	Receive Rule Entry Control Register	RSCANnGAFLECTR	<RSCFDn_base> + H’0098
RSCANn	Receive Rule Configuration Register 0	RSCANnGAFLCFG0	<RSCFDn_base> + H’009C
RSCANn	Receive Rule ID Register j	RSCANnGAFLIDj	<RSCFDn_base> + H’0500 + (H’10 × j)
RSCANn	Receive Rule Mask Register j	RSCANnGAFLMj	<RSCFDn_base> + H’0504 + (H’10 × j)
RSCANn	Receive Rule Pointer 0 Register j	RSCANnGAFLP0_j	<RSCFDn_base> + H’0508 + (H’10 × j)
RSCANn	Receive Rule Pointer 1 Register j	RSCANnGAFLP1_j	<RSCFDn_base> + H’050C + (H’10 × j)
Receive buffer-related registers			
RSCANn	Receive Buffer Number Register	RSCANnRMNB	<RSCFDn_base> + H’00A4
RSCANn	Receive Buffer New Data Register y	RSCANnRMNDy	<RSCFDn_base> + H’00A8 + (H’04 × y)
RSCANn	Receive Buffer ID Register q	RSCANnRMIDq	<RSCFDn_base> + H’0600 + (H’10 × q)
RSCANn	Receive Buffer Pointer Register q	RSCANnRMPTRq	<RSCFDn_base> + H’0604 + (H’10 × q)
RSCANn	Receive Buffer Data Field 0 Register q	RSCANnRMDF0_q	<RSCFDn_base> + H’0608 + (H’10 × q)
RSCANn	Receive Buffer Data Field 1 Register q	RSCANnRMDF1_q	<RSCFDn_base> + H’060C + (H’10 × q)

Table 28.10 Registers (2/3)

Module	Register	Symbol	Address
Receive FIFO buffer-related registers			
RSCANn	Receive FIFO Buffer Configuration and Control Register x	RSCANnRFCCx	<RSCFDn_base> + H’00B8 + (H’04 × x)
RSCANn	Receive FIFO Buffer Status Register x	RSCANnRFSTSx	<RSCFDn_base> + H’00D8 + (H’04 × x)
RSCANn	Receive FIFO Buffer Pointer Control Register x	RSCANnRFPCTRx	<RSCFDn_base> + H’00F8 + (H’04 × x)
RSCANn	Receive FIFO Buffer Access ID Register x	RSCANnRFIDx	<RSCFDn_base> + H’0E00 + (H’10 × x)
RSCANn	Receive FIFO Buffer Access Pointer Register x	RSCANnRFPTRx	<RSCFDn_base> + H’0E04 + (H’10 × x)
RSCANn	Receive FIFO Buffer Access Data Field 0 Register x	RSCANnRFDF0_x	<RSCFDn_base> + H’0E08 + (H’10 × x)
RSCANn	Receive FIFO Buffer Access Data Field 1 Register x	RSCANnRFDF1_x	<RSCFDn_base> + H’0E0C + (H’10 × x)
Transmit/Receive FIFO buffer related registers			
RSCANn	Transmit/receive FIFO Buffer Configuration and Control Register k	RSCANnCFCCk	<RSCFDn_base> + H’0118 + (H’04 × k)
RSCANn	Transmit/receive FIFO Buffer Status Register k	RSCANnCFSTSk	<RSCFDn_base> + H’0178 + (H’04 × k)
RSCANn	Transmit/receive FIFO Buffer Pointer Control Register k	RSCANnCFPCTRk	<RSCFDn_base> + H’01D8 + (H’04 × k)
RSCANn	Transmit/receive FIFO Buffer Access ID Register k	RSCANnCFIDk	<RSCFDn_base> + H’0E80 + (H’10 × k)
RSCANn	Transmit/receive FIFO Buffer Access Pointer Register k	RSCANnCFPTRk	<RSCFDn_base> + H’0E84 + (H’10 × k)
RSCANn	Transmit/receive FIFO Buffer Access Data Field 0 Register k	RSCANnCFDF0_k	<RSCFDn_base> + H’0E88 + (H’10 × k)
RSCANn	Transmit/receive FIFO Buffer Access Data Field 1 Register k	RSCANnCFDF1_k	<RSCFDn_base> + H’0E8C + (H’10 × k)
FIFO status-related registers			
RSCANn	FIFO Empty Status Register	RSCANnFESTS	<RSCFDn_base> + H’0238
RSCANn	FIFO Full Status Register	RSCANnFFSTS	<RSCFDn_base> + H’023C
RSCANn	FIFO Message Lost Status Register	RSCANnFMSTS	<RSCFDn_base> + H’0240
RSCANn	RSCANn FIFO Receive FIFO Buffer Interrupt Flag Status Register	RSCANnRFISTS	<RSCFDn_base> + H’0244
RSCANn	Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register	RSCANnCFRISTS	<RSCFDn_base> + H’0248
RSCANn	Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCANnCFTISTS	<RSCFDn_base> + H’024C
Transmit buffer-related registers			
RSCANn	Transmit Buffer Control Register p	RSCANnTMCp	<RSCFDn_base> + H’0250 + (H’01 × p)
RSCANn	Transmit Buffer Status Register p	RSCANnTMSTSp	<RSCFDn_base> + H’02D0 + (H’01 × p)
RSCANn	Transmit Buffer ID Register p	RSCANnTMIDp	<RSCFDn_base> + H’1000 + (H’10 × p)
RSCANn	Transmit Buffer Pointer Register p	RSCANnTMPTRp	<RSCFDn_base> + H’1004 + (H’10 × p)
RSCANn	Transmit Buffer Data Field 0 Register p	RSCANnTMDF0_p	<RSCFDn_base> + H’1008 + (H’10 × p)
RSCANn	Transmit Buffer Data Field 1 Register p	RSCANnTMDF1_p	<RSCFDn_base> + H’100C + (H’10 × p)
RSCANn	Transmit Buffer Interrupt Enable Configuration Register y	RSCANnTMIECy	<RSCFDn_base> + H’0390 + (H’04 × y)
Transmit buffer status-related registers			
RSCANn	Transmit Buffer Transmit Request Status Register y	RSCANnTMTRSTSy	<RSCFDn_base> + H’0350 + (H’04 × y)
RSCANn	Transmit Buffer Transmit Abort Request Status Register y	RSCANnTMTARSTSy	<RSCFDn_base> + H’0360 + (H’04 × y)
RSCANn	Transmit Buffer Transmit Complete Status Register y	RSCANnTMTCSTSy	<RSCFDn_base> + H’0370 + (H’04 × y)
RSCANn	Transmit Buffer Transmit Abort Status Register y	RSCANnTMTASTSy	<RSCFDn_base> + H’0380 + (H’04 × y)
Transmit queue-related registers			
RSCANn	Transmit Queue Configuration and Control Register m	RSCANnTXQCCm	<RSCFDn_base> + H’03A0 + (H’04 × m)
RSCANn	Transmit Queue Status Register m	RSCANnTXQSTSm	<RSCFDn_base> + H’03C0 + (H’04 × m)
RSCANn	Transmit Queue Pointer Control Register m	RSCANnTXQPCTRm	<RSCFDn_base> + H’03E0 + (H’04 × m)

Table 28.10 Registers (3/3)

Module	Register	Symbol	Address
Transmit history-related registers			
RSCANn	Transmit History Configuration and Control Register m	RSCANnTHLCCm	<RSCFDn_base> + H'0400 + (H'04 × m)
RSCANn	Transmit History Status Register m	RSCANnTHLSTSm	<RSCFDn_base> + H'0420 + (H'04 × m)
RSCANn	Transmit History Pointer Control Register m	RSCANnTHLPCTRm	<RSCFDn_base> + H'0440 + (H'04 × m)
RSCANn	Transmit History Access Register m	RSCANnTHLACCm	<RSCFDn_base> + H'1800 + (H'04 × m)
Test-related registers			
RSCANn	Global Test Configuration Register	RSCANnGTSTCFG	<RSCFDn_base> + H'0468
RSCANn	Global Test Control Register	RSCANnGTSTCTR	<RSCFDn_base> + H'046C
RSCANn	Global Lock Key Register	RSCANnGLOCKK	<RSCFDn_base> + H'047C
RSCANn	RAM Test Page Access Register r	RSCANnRPGACCr	<RSCFDn_base> + H'1900 + (H'04 × r)

Table 28.11 Transmit Buffer p Allocated to Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 28.12 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANm
Transmit/receive FIFO buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 28.13 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000b	Transmit buffer 16 × m + 0
0001b	Transmit buffer 16 × m + 1
0010b	Transmit buffer 16 × m + 2
0011b	Transmit buffer 16 × m + 3
0100b	Transmit buffer 16 × m + 4
0101b	Transmit buffer 16 × m + 5
0110b	Transmit buffer 16 × m + 6
0111b	Transmit buffer 16 × m + 7
1000b	Transmit buffer 16 × m + 8
1001b	Transmit buffer 16 × m + 9
1010b	Transmit buffer 16 × m + 10
1011b	Transmit buffer 16 × m + 11
1100b	Transmit buffer 16 × m + 12
1101b	Transmit buffer 16 × m + 13
1110b	Transmit buffer 16 × m + 14
1111b	Transmit buffer 16 × m + 15

Table 28.14 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC[3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000b	Setting prohibited
0001b	Setting prohibited
0010b	Transmit buffer 16 × m + 15 to 16 × m + 13
0011b	Transmit buffer 16 × m + 15 to 16 × m + 12
0100b	Transmit buffer 16 × m + 15 to 16 × m + 11
0101b	Transmit buffer 16 × m + 15 to 16 × m + 10
0110b	Transmit buffer 16 × m + 15 to 16 × m + 9
0111b	Transmit buffer 16 × m + 15 to 16 × m + 8
1000b	Transmit buffer 16 × m + 15 to 16 × m + 7
1001b	Transmit buffer 16 × m + 15 to 16 × m + 6
1010b	Transmit buffer 16 × m + 15 to 16 × m + 5
1011b	Transmit buffer 16 × m + 15 to 16 × m + 4
1100b	Transmit buffer 16 × m + 15 to 16 × m + 3
1101b	Transmit buffer 16 × m + 15 to 16 × m + 2
1110b	Transmit buffer 16 × m + 15 to 16 × m + 1
1111b	Transmit buffer 16 × m + 15 to 16 × m + 0

28.3.2 Details of Interface Mode-related Registers

28.3.2.1 RSCANnGRMCFG — Global Interface Mode Select Register

Access Size:	RSCANnGRMCFG register can be read/written in 32-bit units RSCANnGRMCFGL, RSCANnGRMCFGHregisters can be read/written in 16-bit units RSCANnGRMCFGLL, RSCANnGRMCFGLH, RSCANnGRMCFGHL, RSCANnGRMCFGHHregisters can be read/written in 8-bit units
Address(es):	RSCANnGRMCFG: <RSCFDn_base> + H'04FC, RSCANnGRMCFGL: <RSCFDn_base> + H'04FC, RSCANnGRMCFGH: <RSCFDn_base> + H'04FE, RSCANnGRMCFGLL: <RSCFDn_base> + H'04FC, RSCANnGRMCFGLH: <RSCFDn_base> + H'04FD, RSCANnGRMCFGHL: <RSCFDn_base> + H'04FE, RSCANnGRMCFGHH: <RSCFDn_base> + H'04FF
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RCMC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
0	RCMC	0	R/W	Interface Mode Select 0: Classical CAN mode 1: CANFD mode

Note: The RSCANnGRMCFG register and the RSCFDnCFDGRMCFG register are the same register. Therefore, set either one of the registers.

Modify the RSCANnGRMCFG register only in global reset mode. Before setting other RS-CANFD registers, set this register.

RCMC Bit

Setting this bit to 0 places the RS-CANFD module in classical CAN mode.

Setting this bit to 1 places the RS-CANFD module in CANFD mode. To switch CANFD mode to classical CAN mode, set the value after reset in all registers and bits allocated to the register map of CANFD mode and then modify the RSCANnGRMCFG register.

28.3.3 Details of Channel-related Registers

28.3.3.1 RSCANnCmCFG — Channel Configuration Register (m = 0, 1)

Access Size:	RSCANnCmCFG register can be read/written in 32-bit units RSCANnCmCFGL, RSCANnCmCFGH registers can be read/written in 16-bit units RSCANnCmCFGLL, RSCANnCmCFGLH, RSCANnCmCFGHL, RSCANnCmCFGHH registers can be read/written in 8-bit unitsa
Address(es):	RSCANnCmCFG: <RSCFDn_base> + H'0000 + (H'10 × m) RSCANnCmCFGL: <RSCFDn_base> + H'0000 + (H'10 × m), RSCANnCmCFGH: <RSCFDn_base> + H'0002 + (H'10 × m) RSCANnCmCFGLL: <RSCFDn_base> + H'0000 + (H'10 × m), RSCANnCmCFGLH: <RSCFDn_base> + H'0001 + (H'10 × m), RSCANnCmCFGHL: <RSCFDn_base> + H'0002 + (H'10 × m), RSCANnCmCFGHH: <RSCFDn_base> + H'0003 + (H'10 × m)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	SJW[1:0]		—	TSEG2[2:0]			TSEG1[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BRP[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
25, 24	SJW[1:0]	All 0	R/W	Resynchronization Jump Width Control b25 b24 0 0: 1 Tq 0 1: 2 Tq 1 0: 3 Tq 1 1: 4 Tq
23	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
22 to 20	TSEG2[2:0]	All 0	R/W	Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq

Bit	Bit Name	Initial Value	R/W	Description
19 to 16	TSEG1[3:0]	All 0	R/W	Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: Setting prohibited 0 0 1 0: Setting prohibited 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 10	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
9 to 0	BRP[9:0]	All 0	R/W	Prescaler Division Ratio Set When these bits are set to P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.

Modify the RSCANnCmCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode before shifting to channel communication mode or channel halt mode. For a description of the bit timing parameters and settings, see **Section 28.11.1, Initial Settings**.

SJW[1:0] Bits

These bits are used to specify a Tq value for the resynchronization jump width.

Allowed values are 1 Tq to 4 Tq, inclusive.

Set a value less than or equal to the value of the TSEG2 bits.

TSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2).

Allowed values are 2 Tq to 8 Tq, inclusive.

Set a value smaller than the value of the TSEG1 bits.

TSEG1[3:0] Bits

These bits are used to specify a Tq value for the total length of the propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1).

Allowed values are 4 Tq to 16 Tq, inclusive.

BRP[9:0] Bits

The CANmTq clock (fCANTQm) is calculated by dividing the CAN clock (fCAN) by the baud rate prescaler, ((BRP[9:0]) + 1). One clock cycle of the CANmTq clock is 1 Time Quantum (Tq).

28.3.3.2 RSCANnCmCTR — Channel Control Register (m = 0, 1)

Access Size:	RSCANnCmCTR register can be read/written in 32-bit units RSCANnCmCTRL, RSCANnCmCTRH registers can be read/written in 16-bit units RSCANnCmCTRLL, RSCANnCmCTRLH, RSCANnCmCTRHL, RSCANnCmCTRHH registers can be read/written in 8-bit units
Address(es):	RSCANnCmCTR: <RSCFDn_base> + H’0004 + (H’10 × m) RSCANnCmCTRL: <RSCFDn_base> + H’0004 + (H’10 × m), RSCANnCmCTRH: <RSCFDn_base> + H’0006 + (H’10 × m) RSCANnCmCTRLL: <RSCFDn_base> + H’0004 + (H’10 × m), RSCANnCmCTRLH: <RSCFDn_base> + H’0005 + (H’10 × m), RSCANnCmCTRHL: <RSCFDn_base> + H’0006 + (H’10 × m), RSCANnCmCTRHH: <RSCFDn_base> + H’0007 + (H’10 × m)
Initial Value:	H’0000 0005

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCT	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	—	—	—	TAIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
30	CRCT	0	R/W	CRC Error Test Enable 0: The first bit of the reception ID field is not inverted. 1: The first bit of the reception ID field is inverted.
29 to 27	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
26, 25	CTMS[1:0]	All 0	R/W	Communication Test Mode Select b26 b25 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	0	R/W	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	0	R/W	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits 14 to 8 in the RSCANnCmERFL register are all cleared. 1: Error flags for all error information are displayed.

Bit	Bit Name	Initial Value	R/W	Description
22, 21	BOM[1:0]	All 0	R/W	Bus Off Recovery Mode Select b22 b21 0 0: ISO11898-1 compliant 0 1: Entry to channel halt mode automatically at bus-off entry 1 0: Entry to channel halt mode automatically at bus-off end 1 1: Entry to channel halt mode (in bus-off state) by program request
20 to 17	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
16	TAIE	0	R/W	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	0	R/W	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	0	R/W	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	0	R/W	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.
12	BORIE	0	R/W	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	0	R/W	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	0	R/W	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	0	R/W	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	0	R/W	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
3	RTBO	0	R/W	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	1	R/W	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	01b	R/W	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

CRCT Bit

This bit is used to test the CRC generation circuit in the RS-CANFD module. Setting this bit to 1 inverts the first bit of the ID field when a message is received. With this inversion of bit, the CRC calculation result does not match the normal CRC value of the received frame, which can detect a CRC error (the CERR bit in the RSCANnCmERFL register is 1). When using this function, note the following.

- This function is available while the CTME bit in the RSCANnCmCTR register is 1 (communication test mode enabled).
- This function cannot communicate with other CAN nodes. Use this function for inter-channel communication test (the CmICBCE bit in the RSCANnGTSTCFG register is 1).
- Bit inversion in the ID field may cause bit stuffing rule violation. In that case, no CRC error is detected but a stuff error is detected.

Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CTMS[1:0] Bit

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify this bit in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control the display mode of bits 14 to 8 in the RSCANnCmERFL register.

When this bit is clear to 0, if any error is detected while the flags of bits 14-8 in the RSCANnCmERFL register are all 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.

When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order. Modify this bit only in channel reset mode or channel halt mode.

BOM[1:0] Bit

These bits are used to select the bus off recovery mode of the RS-CANFD module.

When the BOM[1:0] bits are set to 00b, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10b (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 01b, the CHMDC[1:0] bits in the RSCFDnCmCTRRSCANnCmCTR register (m = 0, 1) are set to 10b and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCFDnCmSTSRSCANnCmSTS register are cleared to H’00.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 10b, the CHMDC[1:0] bits are set to 10b and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H’00.

When the BOM[1:0] bits are set to 11b and the CHMDC[1:0] bits are set to 10b while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H'00. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10b, a bus off recovery interrupt request is generated.

If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01b or at bus off end when the BOM[1:0] bits are 10b), the program's writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.

TAIE Bit

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.

ALIE Bit

When the ALF flag in the RSCANnCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BLIE Bit

When the BLF flag in the RSCANnCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCANnCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BORIE Bit

When the BORF flag in the RSCANnCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCANnCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EPIE Bit

When the EPF flag in the RSCANnCmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EWIE Bit

When the EWF flag in the RSCANnCmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BEIE Bit

When the BEF flag in the RSCANnCmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

RTBO Bit

Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCANnCmSTS register to H’00 and also clears the BOSTS flag in the RSCANnCmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCANnCmCTR register are 00b (ISO11898-1 compliant).

A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RSCAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel into channel stop mode. Clearing this bit to 0 makes the channel exit channel stop mode.

This bit should not be modified in channel communication mode or channel halt mode.

CHMDC[1:0] Bit

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see **Section 28.6.2, Channel Modes**. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to 11b. When the CAN module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become 10b.

28.3.3.3 RSCANnCmSTS — Channel Status Register (m = 0, 1)

Access Size:	RSCANnCmSTS register can be read only in 32-bit units RSCANnCmSTSL, RSCANnCmSTSH registers can be read only in 16-bit units RSCANnCmSTSLL, RSCANnCmSTSLH, RSCANnCmSTSHL, RSCANnCmSTSHH registers can be read only in 8-bit units
Address(es):	RSCANnCmSTS: <RSCFDn_base> + H’0008 + (H’10 × m) RSCANnCmSTSL: <RSCFDn_base> + H’0008 + (H’10 × m), RSCANnCmSTSH: <RSCFDn_base> + H’000A + (H’10 × m) RSCANnCmSTSLL: <RSCFDn_base> + H’0008 + (H’10 × m), RSCANnCmSTSLH: <RSCFDn_base> + H’0009 + (H’10 × m), RSCANnCmSTSHL: <RSCFDn_base> + H’000A + (H’10 × m), RSCANnCmSTSHH: <RSCFDn_base> + H’000B + (H’10 × m)
Initial Value:	H’0000 0005

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TEC[7:0]	All 0	R	The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	All 0	R	The receive error counter (REC) can be read.
15 to 8	—	All 0	R	Reserved These bits are read as the value after reset.
7	COMSTS	0	R	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	0	R	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	0	R	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	0	R	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	0	R	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	1	R	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	0	R	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode

Bit	Bit Name	Initial Value	R/W	Description
0	CRSTSTS	1	R	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

COMSTS Flag

This bit indicates that communication is ready.

This flag becomes 1 when the CAN module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the bus off state (TEC[7:0] > 255) is entered. It is cleared to 0 when the CAN module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the RS-CANFD module has entered the error passive state (($128 \leq$ TEC[7:0] ≤ 255) or ($128 \leq$ REC[7:0])), It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel stop mode, and is cleared to 0 when the CAN module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel halt mode, and is cleared to 0 when the CAN module has returned from channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel reset mode, and is cleared to 0 when the CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the CAN module transitions from channel reset mode to channel stop mode.

28.3.3.4 RSCANnCmERFL — Channel Error Flag Register (m = 0, 1)

Access Size:	RSCANnCmERFL register can be read/written in 32-bit units RSCANnCmERFLL, RSCANnCmERFLH registers can be read/written in 16-bit units RSCANnCmERFLLL, RSCANnCmERFLLH, RSCANnCmERFLHL, RSCANnCmERFLHH registers can be read/written in 8-bit units
Address(es):	RSCANnCmERFL: <RSCFDn_base> + H’000C + (H’10 × m) RSCANnCmERFLL: <RSCFDn_base> + H’000C + (H’10 × m), RSCANnCmERFLH: <RSCFDn_base> + H’000E + (H’10 × m) RSCANnCmERFLLL: <RSCFDn_base> + H’000C + (H’10 × m), RSCANnCmERFLLH: <RSCFDn_base> + H’000D + (H’10 × m), RSCANnCmERFLHL: <RSCFDn_base> + H’000E + (H’10 × m), RSCANnCmERFLHH: <RSCFDn_base> + H’000F + (H’10 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved When read, the value after reset is returned. When writing to this bit, write the value after reset.
30 to 16	CRCREG [14:0]	All 0	R	CRC Calculation Data A CRC value calculated based on the transmit message or receive message is indicated.
15	—	0	R	Reserved When read, the value after reset is returned. When writing to this bit, write the value after reset.
14	ADERR	0	R/W*1	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	0	R/W*1	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	0	R/W*1	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	0	R/W*1	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.
10	AERR	0	R/W*1	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.

Bit	Bit Name	Initial Value	R/W	Description
9	FERR	0	R/W*1	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	0	R/W*1	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	0	R/W*1	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.
6	BLF	0	R/W*1	Bus Lock Flag 0: No channel bus is detected. 1: Channel bus is detected.
5	OVLF	0	R/W*1	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	0	R/W*1	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	0	R/W*1	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	0	R/W*1	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	0	R/W*1	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	0	R/W*1	Bus Error Flag 0: No channel bus error is detected. 1: Channel bus error is detected.

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.

If the ERRD bit in the RSCANnCmCTR register is set to 0 (i.e., only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCANnCmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the when time the error occurred.

CRCREG[14:0] Flag

When the CTME bit in the RSCANnCmCTR register is set to 1 (communication test mode is enabled), the CRC value calculated based on the transmit or receive message can be read. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0. This bit is always 0 in channel reset mode.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration-lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.

- A recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- The CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the CAN module returns from the bus off state. However, this flag is not set to 1 if the CAN module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCANnCmCTR register are set to 01b (channel reset mode).
- The RTBO bit in the RSCANnCmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCANnCmCTR register are set to 01b (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCANnCmCTR register are set to 10b (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11b (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCANnCmCTR register (m = 0, 1) set to 01b (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127).

This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCANnCmERFL register is set to 1.

NOTE

To clear the flag of this register to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

28.3.4 Details of Global-related Registers

28.3.4.1 RSCANnGCFG — Global Configuration Register

Access Size:	RSCANnGCFG register can be read/written in 32-bit units RSCANnGCFGL, RSCANnGCFGH registers can be read/written in 16-bit units RSCANnGCFGLL, RSCANnGCFGLH, RSCANnGCFGHL, RSCANnGCFGHH registers can be read/written in 8-bit units
Address(es):	RSCANnGCFG: <RSCFDn_base> + H’0084 RSCANnGCFGL: <RSCFDn_base> + H’0084, RSCANnGCFGH: <RSCFDn_base> + H’0086 RSCANnGCFGLL: <RSCFDn_base> + H’0084, RSCANnGCFGLH: <RSCFDn_base> + H’0085, RSCANnGCFGHL: <RSCFDn_base> + H’0086, RSCANnGCFGHH: <RSCFDn_base> + H’0087
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				TMTSCE	—	—	DCS	MME	DRE	DCE	TPRI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	ITRCP[15:0]	All 0	R/W	Interval Timer Prescaler Set When these bits are set to M, the pclk is divided by M. Setting H’0000 is prohibited when the interval timer is in use.
15 to 13	TSBTCS[2:0]	All 0	R/W	Timestamp Clock Source Select b15 b14 b13 0 0 0: Channel 0 bit time clock 0 0 1: Channel 1 bit time clock 0 1 0: Setting prohibited 0 1 1: Setting prohibited 1 0 0: Setting prohibited 1 0 1: Setting prohibited 1 1 0: Setting prohibited 1 1 1: Setting prohibited
12	TSSS	0	R/W	Timestamp Source Select 0: pclk/2[*1] 1: Bit time clock

Bit	Bit Name	Initial Value	R/W	Description
11 to 8	TSP[3:0]	All 0	R/W	Timestamp Clock Source Division b11 b10 b9 b8 0 0 0 0: Not divided 0 0 0 1: Divided by 2 0 0 1 0: Divided by 4 0 0 1 1: Divided by 8 0 1 0 0: Divided by 16 0 1 0 1: Divided by 32 0 1 1 0: Divided by 64 0 1 1 1: Divided by 128 1 0 0 0: Divided by 256 1 0 0 1: Divided by 512 1 0 1 0: Divided by 1024 1 0 1 1: Divided by 2048 1 1 0 0: Divided by 4096 1 1 0 1: Divided by 8192 1 1 1 0: Divided by 16384 1 1 1 1: Divided by 32768
7	TMTSCE	0	R/W	Transmission Timestamp Enable 0: Transmission timestamp is disabled. 1: Transmission timestamp is enabled.
6, 5	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
4	DCS	0	R/W	CAN Clock Source Select*[2] 0: clkc 1: clk_xincan
3	MME	0	R/W	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	0	R/W	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	0	R/W	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	0	R/W	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000b.

Note 2. For the CAN clock frequency settings, see **Section 28.1.3, Clock Supply**.

Modify the RSCANnGCFG register only in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. See **Section 28.8.3.1, Interval Transmission Function**.

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the bit time clock that will be the clock source of the timestamp counter.

TSSS Bits

This bit is used to select a clock source of the timestamp counter.

TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

TMTSCE Bits

Setting this bit to 1 makes it possible to store the timestamp of a transmitted message in the transmit history buffer. The timestamp is stored in TMTS[15:0] bits in the RSCANnTHLACCm register.

DCS Bits

When this bit is set to 0, clkc is used as the clock source of the CAN clock (fCAN).

When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN). For the CAN clock frequency settings, see **Table 28.6**.

MME Bits

Setting this bit to 1 makes the mirror function available.

DRE Bits

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of H’00 is stored in each data byte beyond the DLC value of the receive rule.

The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

DCE Bits

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCANnGAFLP0_j register to 0000b before clearing the DCE bit in the RSCANnGCFG register to 0.

TPRI Bits

This bit is used to set the transmit priority.

When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.

While the transmit queue is in use, this bit should be set to 0.

28.3.4.2 RSCANnGCTR — Global Control Register

Access Size:	RSCANnGCTR register can be read/written in 32-bit units RSCANnGCTRL, RSCANnGCTRH registers can be read/written in 16-bit units RSCANnGCTRLL, RSCANnGCTRLH, RSCANnGCTRHL, RSCANnGCTRHH registers can be read/written in 8-bit units
Address(es):	RSCANnGCTR: <RSCFDn_base> + H’0088 RSCANnGCTRL: <RSCFDn_base> + H’0088, RSCANnGCTRH: <RSCFDn_base> + H’008A RSCANnGCTRLL: <RSCFDn_base> + H’0088, RSCANnGCTRLH: <RSCFDn_base> + H’0089, RSCANnGCTRHL: <RSCFDn_base> + H’008A, RSCANnGCTRHH: <RSCFDn_base> + H’008B
Initial Value:	H’0000 0005

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
16	TSRST	0	R/W	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 11	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
10	THLEIE	0	R/W	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	0	R/W	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	0	R/W	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	GSLPR	1	R/W	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	01b	R/W	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCANnGTSCregister is cleared to H’0000.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in the RSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in the RSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

GSLPR Bit

Setting this bit to 1 places the RSCAN module into global stop mode.

Clearing this bit to 0 makes the RSCAN module leave from global stop mode.

This bit should not be modified in global operating mode or global test mode.

GMDC[1:0] Bit

These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see **Section 28.6.1, Global Modes**. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.

28.3.4.3 RSCANnGSTS — Global Status Register

Access Size:	RSCANnGSTS register can be read only in 32-bit units RSCANnGSTSL, RSCANnGSTSH registers can be read only in 16-bit units RSCANnGSTSLL, RSCANnGSTSLH, RSCANnGSTSHL, RSCANnGSTSHH registers can be read only in 8-bit units
Address(es):	RSCANnGSTS: <RSCFDn_base> + H'008C RSCANnGSTSL: <RSCFDn_base> + H'008C, RSCANnGSTSH: <RSCFDn_base> + H'008E RSCANnGSTSLL: <RSCFDn_base> + H'008C, RSCANnGSTSLH: <RSCFDn_base> + 008D, RSCANnGSTSHL: <RSCFDn_base> + H'008E, RSCANnGSTSHH: <RSCFDn_base> + H'008F
Initial Value:	H'0000 000D

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
3	GRAMINIT	1	R	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	1	R	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	0	R	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	1	R	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.

This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to global stop mode, and is cleared to 0 when the CAN module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global test mode, and is cleared to 0 when the CAN module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global reset mode, and is cleared to 0 when the CAN module has exited global reset mode. This flag remains 1 even when the CAN module has transitioned from global reset mode to global stop mode.

28.3.4.4 RSCANnGERFL — Global Error Flag Register

Access Size: RSCANnGERFL register can be read/written in 32-bit units

RSCANnGERFLL, RSCANnGERFLH registers can be read/written in 16-bit units

RSCANnGERFLLL, RSCANnGERFLLH, RSCANnGERFLHL, RSCANnGERFLHH registers can be read/written in 8-bit units

Address(es): RSCANnGERFL: <RSCFDn_base> + H’0090

RSCANnGERFLL: <RSCFDn_base> + H’0090, RSCANnGERFLH: <RSCFDn_base> + H’0092

RSCANnGERFLLL: <RSCFDn_base> + H’0090, RSCANnGERFLLH: <RSCFDn_base> + H’0091,

RSCANnGERFLHL: <RSCFDn_base> + H’0092, RSCANnGERFLHH: <RSCFDn_base> + H’0093

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	THLES	MES	DEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*[1]

Bit	Bit Name	Initial Value	R/W	Description
31 to 14, 7, 6, 4, 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
13 to 8, 5	—	All 0	R	Reserved When read, an undefined value is returned. The write value should be the value after reset.
2	THLES	0	R	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	0	R	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	0	R/W*[1]	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

All flags in the RSCANnGERFL register are cleared to 0 in global reset mode.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCANnTHLSTSm register (m = 0, 1) is set to 1. This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCANnRFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCANnCFSTSk register (k = 0 to 5) is set to 1.

This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.

NOTE

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”

28.3.4.5 RSCANnGTSC — Global Timestamp Counter Register

Access Size:	RSCANnGTSC register can be read only in 32-bit units. RSCANnGTSCL, RSCANnGTSCH registers can be read only in 16-bit units.
Address(es):	RSCANnGTSC: <RSCFDn_base> + H'0094 RSCANnGTSCL: <RSCFDn_base> + H'0094, RSCANnGTSCH: <RSCFDn_base> + H'0096
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset.
15 to 0	TS[15:0]	All 0	R	Timestamp Value The timestamp counter value can be read. Counter Value: H'0000 to H'FFFF

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. Furthermore, while the TMTSCE bit in the RSCANnGCFG register is 1, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.

The timestamp counter start timing and stop timing depend on the count source.

- When the TSSS bit in the GRSCANnGCFG register is 0 (pclk):
  The timestamp counter starts counting when the RSCAN module has transitioned to global operating mode. This counter stops counting when the RSCAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.
  This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

28.3.4.6 RSCANnGTINTSTS0 — Global TX Interrupt Status Register 0

Access Size:	RSCANnGTINTSTS0 register can be read only in 32-bit units RSCANnGTINTSTS0L, RSCANnGTINTSTS0H registers can be read only in 16-bit units RSCANnGTINTSTS0LL, RSCANnGTINTSTS0LH, RSCANnGTINTSTS0HL, RSCANnGTINTSTS0HH registers can be read only in 8-bit units
Address(es):	RSCANnGTINTSTS0: <RSCFDn_base> + H'0460 RSCANnGTINTSTS0L: <RSCFDn_base> + H'0460, RSCANnGTINTSTS0H: <RSCFDn_base> + H'0462 RSCANnGTINTSTS0LL: <RSCFDn_base> + H'0460, RSCANnGTINTSTS0LH: <RSCFDn_base> + H'0461, RSCANnGTINTSTS0HL: <RSCFDn_base> + H'0462, RSCANnGTINTSTS0HH: <RSCFDn_base> + H'0463
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are read as the value after reset.
12	THIF1	0	R*1	Channel 1 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	0	R*1	Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
10	TQIF1	0	R*1	Channel 1 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	0	R*1	Channel 1 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
8	TSIF1	0	R*1	Channel 1 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset.
4	THIF0	0	R*1	Channel 0 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	0	R*1	Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
2	TQIF0	0	R*1	Channel 0 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.

Bit	Bit Name	Initial Value	R/W	Description
1	TAIF0	0	R*[1]	Channel 0 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
0	TSIF0	0	R*[1]	Channel 0 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

TSIFm Bits

The TSIFm bit is set to 1 when the TMIEp bit in the RSCANnTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCANnTMSTSp register are set to 10b (transmit completed without abort request) or 11b (transmit completed with abort request).

When the TMTRF[1:0] flags are cleared to 00b under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.

TAIFm Bits

The TAIFm bit is set to 1 when the TAIE bit in the RSCANnCmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCANnTMSTSp register are set to 01b (transmit abort completed).

This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00b after the transmit abort is completed.

TQIFm Bits

When the TXQIE bit in the RSCANnTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCANnTXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.

When the TXQIF bit (transmit queue interrupt request) in the RSCANnTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

CFTIFm Bits

When the CFTXIE bit in the RSCANnCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCANnCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.

When the CFTXIF bit is cleared to 0 under the conditions that the CFTIFm bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

THIFm Bits

When the THLIE bit in the RSCANnTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCANnTHLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.

When the THLIF bit in the RSCANnTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

28.3.4.7 RSCANnGFDCFG — Global FD configuration register

Access Size:	RSCANnGFDCFG register can be read/written in 32-bit unit RSCANnGFDCFGL and RSCANnGFDCFGH registers can be read/written in 16-bit unit RSCANnGFDCFGLL, RSCANnGFDCFGLH, RSCANnGFDCFGHL, RSCANnGFDCFGHH registers can be read/written in 8-bit unit
Address(es):	RSCANnGFDCFG: <RSCFDn_base> + H'0474 RSCANnGFDCFG L: <RSCFDn_base> + H'0474, RSCANnGFDCFG H: <RSCFDn_base> + H'0476 RSCANnGFDCFG LL: <RSCFDn_base> + H'0474, RSCANnGFDCFG LH: <RSCFDn_base> + H'0475, RSCANnGFDCFG HL: <RSCFDn_base> + H'0476, RSCANnGFDCFG HH: <RSCFDn_base> + H'0477
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSCCFG[1:0]		—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R/W	Reserved These bits are read as the value after reset. The write value should be the value after reset.
9, 8	TSCCFG[1:0]	All 0	R/W	Time-stamp capture setting bit b9 b8 0 0: Captured at a sample point in the SOF bit. 0 1: Captured when a valid frame has been transmitted/received. 1 0: Setting prohibited 1 1: Setting prohibited
7 to 1	—	All 0	R/W	Reserved These bits are read as the value after reset. The write value should be the value after reset.
0	—	0	R/W	Reserved When read, an undefined value is returned. The write value should be the value after reset.

TSCCFG bit

Select a point where a time-stamp value is captured. Modify this bit only in global reset mode.

28.3.5 Details of Receive Rule-related Registers

28.3.5.1 RSCANnGAFLECTR — Receive Rule Entry Control Register

Access Size:	RSCANnGAFLECTR register can be read/written in 32-bit units RSCANnGAFLECTRL, RSCANnGAFLECTRH registers can be read/written in 16-bit units RSCANnGAFLECTRLL, RSCANnGAFLECTRLH, RSCANnGAFLECTRHL, RSCANnGAFLECTRHH registers can be read/written in 8-bit units
Address(es):	RSCANnGAFLECTR: <RSCFDn_base> + H’0098 RSCANnGAFLECTRL: <RSCFDn_base> + H’0098, RSCANnGAFLECTRH: <RSCFDn_base> + H’009A RSCANnGAFLECTRLL: <RSCFDn_base> + H’0098, RSCANnGAFLECTRLH: <RSCFDn_base> + H’0099, RSCANnGAFLECTRHL: <RSCFDn_base> + H’009A, RSCANnGAFLECTRHH: <RSCFDn_base> + H’009B
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
8	AFLDAE	0	R/W	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
4 to 0	AFLPN[4:0]	All 0	R/W	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000b) to page 7 (00111b).

AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.

Set the AFLDAE bit to 1 only in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page. Set these bits to a value within the range of 00000b to 00111b.

28.3.5.2 RSCANnGAFLCFG0 — Receive Rule Configuration Register 0

Access Size:	RSCANnGAFLCFG0 register can be read/written in 32-bit units RSCANnGAFLCFG0L, RSCANnGAFLCFG0H registers can be read/written in 16-bit units RSCANnGAFLCFG0LL, RSCANnGAFLCFG0LH, RSCANnGAFLCFG0HL, RSCANnGAFLCFG0HH registers can be read/written in 8- bit units
Address(es):	RSCANnGAFLCFG0: <RSCFDn_base> + H’009C RSCANnGAFLCFG0L: <RSCFDn_base> + H’009C, RSCANnGAFLCFG0H: <RSCFDn_base> + H’009E RSCANnGAFLCFG0LL: <RSCFDn_base> + H’009C, RSCANnGAFLCFG0LH: <RSCFDn_base> + H’009D, RSCANnGAFLCFG0HL: <RSCFDn_base> + H’009E, RSCANnGAFLCFG0HH: <RSCFDn_base> + H’009F
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RNC0[7:0]	All 0	R/W	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	All 0	R/W	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 0	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCANnGAFLCFG0 register only in global reset mode.

Up to 64 × (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table. Set these bits to a value within the range of H’00 to H’80.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table. Set these bits to a value within the range of H’00 to H’80.

28.3.5.3 RSCANnGAFLIDj — Receive Rule ID Register (j = 0 to 15)

Access Size:	RSCANnGAFLIDj register can be read/written in 32-bit units RSCANnGAFLIDjL, RSCANnGAFLIDjH registers can be read/written in 16-bit units RSCANnGAFLIDjLL, RSCANnGAFLIDjLH, RSCANnGAFLIDjHL, RSCANnGAFLIDjHH registers can be read/written in 8-bit units
Address(es):	RSCANnGAFLIDj: <RSCFDn_base> + H’0500 + (H’10 × j) RSCANnGAFLIDjL: <RSCFDn_base> + H’0500 + (H’10 × j), RSCANnGAFLIDjH: <RSCFDn_base> + H’0502 + (H’10 × j) RSCANnGAFLIDjLL: <RSCFDn_base> + H’0500 + (H’10 × j), RSCANnGAFLIDjLH: <RSCFDn_base> + H’0501 + (H’10 × j), RSCANnGAFLIDjHL: <RSCFDn_base> + H’0502 + (H’10 × j), RSCANnGAFLIDjHH: <RSCFDn_base> + H’0503 + (H’10 × j)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDE	0	R/W	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	0	R/W	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	0	R/W	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
28 to 0	GAFLID[28:0]	All 0	R/W	ID Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCANnGAFLIDj register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.

When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

28.3.5.4 RSCANnGAFLMj — Receive Rule Mask Register (j = 0 to 15)

Access Size:	RSCANnGAFLMj register can be read/written in 32-bit units RSCANnGAFLMjL, RSCANnGAFLMjH registers can be read/written in 16-bit units RSCANnGAFLMjLL, RSCANnGAFLMjLH, RSCANnGAFLMjHL, RSCANnGAFLMjHH registers can be read/written in 8-bit units
Address(es):	RSCANnGAFLMj: <RSCFDn_base> + H’0504 + (H’10 × j) RSCANnGAFLMjL: <RSCFDn_base> + H’0504 + (H’10 × j), RSCANnGAFLMjH: <RSCFDn_base> + H’0506 + (H’10 × j) RSCANnGAFLMjLL: <RSCFDn_base> + H’0504 + (H’10 × j), RSCANnGAFLMjLH: <RSCFDn_base> + H’0505 + (H’10 × j), RSCANnGAFLMjHL: <RSCFDn_base> + H’0506 + (H’10 × j), RSCANnGAFLMjHH: <RSCFDn_base> + H’0507 + (H’10 × j)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDEM	GAFLRTRM	—	GAFLIDM[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDEM	R/W	R/W	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	R/W	R/W	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared
29	—	0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
28 to 0	GAFLIDM [28:0]	All 0	R/W	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCANnGAFLMj register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCANnGAFLIDj register.

When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

28.3.5.5 RSCANnGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access Size: RSCANnGAFLP0_j register can be read/written in 32-bit units
RSCANnGAFLP0_jL, RSCANnGAFLP0_jH registers can be read/written in 16-bit units
RSCANnGAFLP0_jLL, RSCANnGAFLP0_jLH, RSCANnGAFLP0_jHL, RSCANnGAFLP0_jHH registers can be read/written in 8-bit units

Address(es): RSCANnGAFLP0_j: <RSCFDn_base> + H’0508 + (H’10 × j)
RSCANnGAFLP0_jL: <RSCFDn_base> + H’0508 + (H’10 × j),
RSCANnGAFLP0_jH: <RSCFDn_base> + H’050A + (H’10 × j)
RSCANnGAFLP0_jLL: <RSCFDn_base> + H’0508 + (H’10 × j),
RSCANnGAFLP0_jLH: <RSCFDn_base> + H’0509 + (H’10 × j),
RSCANnGAFLP0_jHL: <RSCFDn_base> + H’050A + (H’10 × j),
RSCANnGAFLP0_jHH: <RSCFDn_base> + H’050B + (H’10 × j)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLRMV	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GAFLDLC [3:0]	All 0	R/W	Receive Rule DLC b31 b30 b29 b28 0 0 0 0: DLC check is disabled. 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	GAFLPTR [11:0]	All 0	R/W	Receive Rule Label Set the 12-bit label information.
15	GAFLRMV	0	R/W	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP [6:0]	All 0	R/W	Receive Buffer Number Select Set the receive buffer number to store receive messages.
7 to 0	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCANnGAFLP0_j register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000b disables the DLC check function allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCANnRMNB register.

28.3.5.6 RSCANnGAFLP1_j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access Size:	RSCANnGAFLP1_j register can be read/written in 32-bit units RSCANnGAFLP1_jL, RSCANnGAFLP1_jH registers can be read/written in 16-bit units RSCANnGAFLP1_jLL, RSCANnGAFLP1_jLH, RSCANnGAFLP1_jHL, RSCANnGAFLP1_jHH registers can be read/written in 8-bit units
Address(es):	RSCANnGAFLP1_j: <RSCFDn_base> + H'050C + (H'10 × j) RSCANnGAFLP1_jL: <RSCFDn_base> + H'050C + (H'10 × j), RSCANnGAFLP1_jH: <RSCFDn_base> + H'050E + (H'10 × j) RSCANnGAFLP1_jLL: <RSCFDn_base> + H'050C + (H'10 × j), RSCANnGAFLP1_jLH: <RSCFDn_base> + H'050D + (H'10 × j), RSCANnGAFLP1_jHL: <RSCFDn_base> + H'050E + (H'10 × j), RSCANnGAFLP1_jHH: <RSCFDn_base> + H'050F + (H'10 × j)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	GAFLFDP[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
13 to 8	GAFLFDP [13:8]	All 0	R/W	Transmit/Receive FIFO Buffer k Select (Bit position –8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected.
7 to 0	GAFLFDP [7:0]	All 0	R/W	Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected.

Modify the RSCANnGAFLP1_j register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP[13:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCANnGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/ receive FIFO buffer for which the CFM[1:0] bits in the RSCANnCFCCk register are set to 00b (receive mode) or 10b (gateway mode) are selectable.

28.3.6 Details of Receive Buffer-related Registers

28.3.6.1 RSCANnRMNB — Receive Buffer Number Register

Access Size:	RSCANnRMNB register can be read/written in 32-bit units RSCANnRMNBL, RSCANnRMNBH registers can be read/written in 16-bit units RSCANnRMNBLL, RSCANnRMNBLH, RSCANnRMNBHL, RSCANnRMNBHH registers can be read/written in 8-bit units
Address(es):	RSCANnRMNB: <RSCFDn_base> + H’00A4 RSCANnRMNBL: <RSCFDn_base> + H’00A4, RSCANnRMNBH: <RSCFDn_base> + H’00A6 RSCANnRMNBLL: <RSCFDn_base> + H’00A4, RSCANnRMNBLH: <RSCFDn_base> + H’00A5, RSCANnRMNBHL: <RSCFDn_base> + H’00A6, RSCANnRMNBHH: <RSCFDn_base> + H’00A7
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	NRXMB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
7 to 0	NRXMB[7:0]	All 0	R/W	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 32.

Modify the RSCANnRMNB register only in global reset mode.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels).

Setting these bits all to 0 makes receive buffers unavailable.

28.3.6.2 RSCANnRMNDy — Receive Buffer New Data Register (y = 0)

Access Size:	RSCANnRMNDy register can be read/written in 32-bit units RSCANnRMNDyL, RSCANnRMNDyH registers can be read/written in 16-bit units RSCANnRMNDyLL, RSCANnRMNDyLH, RSCANnRMNDyHL, RSCANnRMNDyHH registers can be read/written in 8-bit units
Address(es):	RSCANnRMNDy: <RSCFDn_base> + H'00A8 + (H'04 × y) RSCANnRMNDyL: <RSCFDn_base> + H'00A8 + (H'04 × y), RSCANnRMNDyH: <RSCFDn_base> + H'00AA + (H'04 × y) RSCANnRMNDyLL: <RSCFDn_base> + H'00A8 + (H'04 × y), RSCANnRMNDyLH: <RSCFDn_base> + H'00A9 + (H'04 × y), RSCANnRMNDyHL: <RSCFDn_base> + H'00AA + (H'04 × y), RSCANnRMNDyHH: <RSCFDn_base> + H'00AB + (H'04 × y)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	RMNSq	All 0	R/W	Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	All 0	R/W	Receive Buffer Receive Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCANnRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 31)

Each RMNSq flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write "0" to the flag and "1" to other flags. These bits cannot be set to 0 while a message is being stored. It takes ten clock cycles of pclk to store a message. These flags are cleared to 0 in global reset mode.

28.3.6.3 RSCANnRMIDq — Receive Buffer ID Register (q = 0 to 31)

Access Size:	RSCANnRMIDq register can be read only in 32-bit units RSCFDnRMIDqRSCANnRMIDqL, RSCFDnRMIDqRSCANnRMIDqH registers can be read only in 16-bit units RSCFDnRMIDqRSCANnRMIDqLL, RSCFDnRMIDqRSCANnRMIDqLH, RSCFDnRMIDqRSCANnRMIDqHL, RSCFDnRMIDqRSCANnRMIDqHH registers can be read only in 8-bit units
Address(es):	RSCANnRMIDq: <RSCFDn_base> + H’0600 + (H’10 × q) RSCANnRMIDqL: <RSCFDn_base> + H’0600 + (H’10 × q), RSCANnRMIDqH: <RSCFDn_base> + H’0602 + (H’10 × q) RSCANnRMIDqLL: <RSCFDn_base> + H’0600 + (H’10 × q), RSCANnRMIDqLH: <RSCFDn_base> + H’0601 + (H’10 × q), RSCANnRMIDqHL: <RSCFDn_base> + H’0602 + (H’10 × q), RSCANnRMIDqHH: <RSCFDn_base> + H’0603 + (H’10 × q)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RMIDE	0	R	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	0	R	Receive Buffer RTR 0: Data frame 1: Remote frame
29	—	0	R	Reserved This bit is read as the value after reset.
28 to 0	RMID[28:0]	All 0	R	Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer

RMRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer.

RMID[28:0] Bits

These bits contain the ID of the message stored in the receive buffer.

28.3.6.4 RSCANnRMPTRq — Receive Buffer Pointer Register (q = 0 to 31)

Access Size:	RSCANnRMPTRq register can be read only in 32-bit units RSCFDnRMPTRqRSCANnRMPTRqL, RSCFDnRMPTRqRSCANnRMPTRqH registers can be read only in 16-bit units RSCFDnRMPTRqRSCANnRMPTRqLL, RSCFDnRMPTRqRSCANnRMPTRqLH, RSCFDnRMPTRqRSCANnRMPTRqHL, RSCFDnRMPTRqRSCANnRMPTRqHH registers can be read only in 8-bit units
Address(es):	RSCANnRMPTRq: <RSCFDn_base> + H'0604 + (H'10 × q) RSCANnRMPTRqL: <RSCFDn_base> + H'0604 + (H'10 × q), RSCANnRMPTRqH: <RSCFDn_base> + H'0606 + (H'10 × q) RSCANnRMPTRqLL: <RSCFDn_base> + H'0604 + (H'10 × q), RSCANnRMPTRqLH: <RSCFDn_base> + H'0605 + (H'10 × q), RSCANnRMPTRqHL: <RSCFDn_base> + H'0606 + (H'10 × q), RSCANnRMPTRqHH: <RSCFDn_base> + H'0607 + (H'10 × q)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RMDLC[3:0]	All 0	R	Receive Buffer DLC Data b31 b30 b29 b28 0 0 0 0: No data byte 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RMPTR[11:0]	All 0	R	Receive Buffer Label Data Label information of the received message.
15 to 0	RMTS[15:0]	All 0	R	Receive Buffer Timestamp Data Timestamp value of the received message.

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

28.3.6.5 RSCANnRMDF0_q — Receive Buffer Data Field 0 Register (q = 0 to 31)

Access Size: RSCANnRMDF0_q register can be read only in 32-bit units
RSCANnRMDF0_qL, RSCANnRMDF0_qH registers can be read only in 16-bit units
RSCANnRMDF0_qLL, RSCANnRMDF0_qLH, RSCANnRMDF0_qHL, RSCANnRMDF0_qHH registers can be read only in 8-bit units

Address(es): RSCANnRMDF0_q: <RSCFDn_base> + H’0608 + (H’10 × q)
RSCANnRMDF0_qL: <RSCFDn_base> + H’0608 + (H’10 × q),
RSCANnRMDF0_qH: <RSCFDn_base> + H’060A + (H’10 × q)
RSCANnRMDF0_qLL: <RSCFDn_base> + H’0608 + (H’10 × q),
RSCANnRMDF0_qLH: <RSCFDn_base> + H’0609 + (H’10 × q),
RSCANnRMDF0_qHL: <RSCFDn_base> + H’060A + (H’10 × q),
RSCANnRMDF0_qHH: <RSCFDn_base> + H’060B + (H’10 × q)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB3[7:0]								RMDB2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB1[7:0]								RMDB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RMDB3[7:0]	All 0	R	Receive Buffer Data Byte 3
23 to 16	RMDB2[7:0]	All 0	R	Receive Buffer Data Byte 2
15 to 8	RMDB1[7:0]	All 0	R	Receive Buffer Data Byte 1
7 to 0	RMDB0[7:0]	All 0	R	Receive Buffer Data Byte 0 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCANnRMPTRq register is smaller than 1000b, data bytes for which no data is set are read as H’00.

28.3.6.6 RSCANnRMDF1_q — Receive Buffer Data Field 1 Register (q = 0 to 31)

Access Size:	RSCANnRMDF1_q register can be read only in 32-bit units RSCANnRMDF1_qL, RSCANnRMDF1_qH registers can be read only in 16-bit units RSCANnRMDF1_qLL, RSCANnRMDF1_qLH, RSCANnRMDF1_qHL, RSCANnRMDF1_qHH registers can be read only in 8-bit units
Address(es):	RSCANnRMDF1_q: <RSCFDn_base> + H’060C + (H’10 × q) RSCANnRMDF1_qL: <RSCFDn_base> + H’060C + (H’10 × q), RSCANnRMDF1_qH: <RSCFDn_base> + H’060E + (H’10 × q) RSCANnRMDF1_qLL: <RSCFDn_base> + H’060C + (H’10 × q), RSCANnRMDF1_qLH: <RSCFDn_base> + H’060D + (H’10 × q), RSCANnRMDF1_qHL: <RSCFDn_base> + H’060E + (H’10 × q), RSCANnRMDF1_qHH: <RSCFDn_base> + H’060F + (H’10 × q)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB7[7:0]								RMDB6[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB5[7:0]								RMDB4[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RMDB7[7:0]	All 0	R	Receive Buffer Data Byte 7 Receive Buffer Data Byte 6 Receive Buffer Data Byte 5 Receive Buffer Data Byte 4 Data for a message stored in the receive buffer can be read.
23 to 16	RMDB6[7:0]	All 0	R	
15 to 8	RMDB5[7:0]	All 0	R	
7 to 0	RMDB4[7:0]	All 0	R	

When the RMDLC[3:0] value in the RSCANnRMPTRq register is smaller than 1000b, data bytes for which no data is set are read as H’00.

28.3.7 Details of Receive FIFO Buffer-related Registers

28.3.7.1 RSCANnRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

Access Size:	RSCANnRFCCxregister can be read/written in 32-bit units RSCANnRFCCxL, RSCANnRFCCxH registers can be read/written in 16-bit units RSCANnRFCCxLL, RSCANnRFCCxLH, RSCANnRFCCxHL, RSCANnRFCCxHH registers can be read/written in 8-bit units
Address(es):	RSCANnRFCCx: <RSCFDn_base> + H'00B8 + (H'04 × x) RSCANnRFCCxL: <RSCFDn_base> + H'00B8 + (H'04 × x), RSCANnRFCCxH: <RSCFDn_base> + H'00BA + (H'04 × x) RSCANnRFCCxLL: <RSCFDn_base> + H'00B8 + (H'04 × x), RSCANnRFCCxLH: <RSCFDn_base> + H'00B9 + (H'04 × x), RSCANnRFCCxHL: <RSCFDn_base> + H'00BA + (H'04 × x), RSCANnRFCCxHH: <RSCFDn_base> + 0 H'0BB + (H'04 × x)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	—	—	—	—	—	RFIE	RFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
15 to 13	RFIGCV[2:0]	All 0	R/W	Receive FIFO Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	0	R/W	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	—	0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	RFDC[2:0]	All 0	R/W	Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 2	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
1	RFIE	0	R/W	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.
0	RFE	0	R/W	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]).

When the RFDC[2:0] bits are set to 001b (4 messages), set the RFIGCV[2:0] bits to 001b, 011b, 101b, or 111b. Modify these bits only in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000b, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCFDnRFSTSxRSCANnRFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode.

Set this bit to 1 with another instruction after the settings to all bits in the RSCANnRFCCx register have been done. This bit is cleared to 0 in global reset mode.

28.3.7.2 RSCANnRFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7)

Access Size:	RSCANnRFSTSx register can be read/written in 32-bit units RSCANnRFSTSxL, RSCANnRFSTSxH registers can be read/written in 16-bit units RSCANnRFSTSxLL, RSCANnRFSTSxLH, RSCANnRFSTSxHL, RSCANnRFSTSxHH registers can be read/written in 8-bit units
Address(es):	RSCANnRFSTSx: <RSCFDn_base> + H’00D8 + (H’04 × x) RSCANnRFSTSxL: <RSCFDn_base> + H’00D8 + (H’04 × x), RSCANnRFSTSxH: <RSCFDn_base> + H’00DA + (H’04 × x) RSCANnRFSTSxLL: <RSCFDn_base> + H’00D8 + (H’04 × x), RSCANnRFSTSxLH: <RSCFDn_base> + H’00D9 + (H’04 × x), RSCANnRFSTSxHL: <RSCFDn_base> + H’00DA + (H’04 × x), RSCANnRFSTSxHH: <RSCFDn_base> + H’00DB + (H’04 × x)
Initial Value:	H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W^{*1}	R/W^{*1}	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	RFMC[7:0]	All 0	R	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
3	RFIF	0	R/W^{*1}	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	0	R/W^{*1}	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	0	R	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	0	R	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

RFMC[7:0] Flag

This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes H’00 when the RFE bit in the RSCANnRFCCx register is set to 0.

RFIF Flag

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCANnRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFMLT Flag

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.

This flag is cleared to 0 in global reset mode or by writing 0 to this flag.

Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCANnRFCCx register.

If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCANnRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCANnRFCCx register is 0 or in global reset mode.

This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

NOTE

To clear the RFMLT or RFIF flag to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

28.3.7.3 RSCANnRFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access Size:	RSCANnRFPCTRx register can only be written in 32-bit units RSCANnRFPCTRxL, RSCANnRFPCTRxH registers can only be written in 16-bit units RSCANnRFPCTRxLL, RSCANnRFPCTRxLH, RSCANnRFPCTRxHL, RSCANnRFPCTRxHH registers can only be written in 8-bit units
Address(es):	RSCANnRFPCTRx: <RSCFDn_base> + H’00F8 + (H’04 × x) RSCANnRFPCTRxL: <RSCFDn_base> + H’00F8 + (H’04 × x), RSCANnRFPCTRxH: <RSCFDn_base> + H’00FA + (H’04 × x) RSCANnRFPCTRxLL: <RSCFDn_base> + H’00F8 + (H’04 × x), RSCANnRFPCTRxLH: <RSCFDn_base> + H’00F9 + (H’04 × x), RSCANnRFPCTRxHL: <RSCFDn_base> + H’00FA + (H’04 × x), RSCANnRFPCTRxHH: <RSCFDn_base> + H’00FB + (H’04 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should be the value after reset.
7 to 0	RFPC[7:0]	All 0	W	Receive FIFO Pointer Control When these bits are set to H’FF, the read pointer moves to the next unread message in the receive FIFO buffer.

RFPC[7:0] Bits

When the RFPC[7:0] bits are set to H’FF, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCANnRFSTSx register is decremented. Read the RSCANnRFIDx, RSCANnRFPTRx, RSCANnRFDF0_x, and RSCANnRFDF1_x registers to read messages in the receive FIFO buffer, and then write H’FF to the RFPC[7:0] bits.

When writing H’FF to these bits, make sure that the RFE bit in the RSCANnRFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCANnRFSTSx register is 0 (the receive FIFO buffer contains unread messages).

28.3.7.4 RSCANnRFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access Size:	RSCANnRFIDx register can be read only in 32-bit units RSCANnRFIDxL, RSCANnRFIDxH registers can be read only in 16-bit units RSCANnRFIDxLL, RSCANnRFIDxLH, RSCANnRFIDxHL, RSCANnRFIDxHH registers can be read only in 8-bit units
Address(es):	RSCANnRFIDx: <RSCFDn_base> + H’0E00 + (H’10 × x) RSCANnRFIDxL: <RSCFDn_base> + H’0E00 + (H’10 × x), RSCANnRFIDxH: <RSCFDn_base> + H’0E02 + (H’10 × x) RSCANnRFIDxLL: <RSCFDn_base> + H’0E00 + (H’10 × x), RSCANnRFIDxLH: <RSCFDn_base> + H’0E01 + (H’10 × x), RSCANnRFIDxHL: <RSCFDn_base> + H’0E02 + (H’10 × x), RSCANnRFIDxHH: <RSCFDn_base> + H’0E03 + (H’10 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFIDE	0	R	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	0	R	Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	—	0	R	Reserved This bit is read as the value after reset.
28 to 0	RFID[28:0]	All 0	R	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

28.3.7.5 RSCANnRFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access Size:	RSCANnRFPTRx register can be read only in 32-bit units RSCANnRFPTRxL, RSCANnRFPTRxH registers can be read only in 16-bit units RSCANnRFPTRxLL, RSCANnRFPTRxLH, RSCANnRFPTRxHL, RSCANnRFPTRxHH registers can be read only in 8-bit units
Address(es):	RSCANnRFPTRx: <RSCFDn_base> + H’0E04 + (H’10 × x) RSCANnRFPTRxL: <RSCFDn_base> + H’0E04 + (H’10 × x), RSCANnRFPTRxH: <RSCFDn_base> + H’0E06 + (H’10 × x) RSCANnRFPTRxLL: <RSCFDn_base> + H’0E04 + (H’10 × x), RSCANnRFPTRxLH: <RSCFDn_base> + H’0E05 + (H’10 × x), RSCANnRFPTRxHL: <RSCFDn_base> + H’0E06 + (H’10 × x), RSCANnRFPTRxHH: <RSCFDn_base> + H’0E07 + (H’10 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RFDLC[3:0]	All 0	R	Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RFPTR[11:0]	All 0	R	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS[15:0]	All 0	R	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

28.3.7.6 RSCANnRFDF0_x — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)

Access Size:	RSCANnRFDF0_x register can be read only in 32-bit units RSCANnRFDF0_xL, RSCANnRFDF0_xH registers can be read only in 16-bit units RSCANnRFDF0_xLL, RSCANnRFDF0_xLH, RSCANnRFDF0_xHL, RSCANnRFDF0_xHH registers can be read only in 8-bit units
Address(es):	RSCANnRFDF0_x: <RSCFDn_base> + H’0E08 + (H’10 × x) RSCANnRFDF0_xL: <RSCFDn_base> + H’0E08 + (H’10 × x), RSCANnRFDF0_xH: <RSCFDn_base> + H’0E0A + (H’10 × x) RSCANnRFDF0_xLL: <RSCFDn_base> + H’0E08 + (H’10 × x), RSCANnRFDF0_xLH: <RSCFDn_base> + H’0E09 + (H’10 × x), RSCANnRFDF0_xHL: <RSCFDn_base> + H’0E0A + (H’10 × x), RSCANnRFDF0_xHH: <RSCFDn_base> + H’0E0B + (H’10 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB3[7:0]								RFDB2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB1[7:0]								RFDB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RFDB3[7:0]	All 0	R	Receive FIFO Buffer Data Byte 3 Receive FIFO Buffer Data Byte 2 Receive FIFO Buffer Data Byte 1 Receive FIFO Buffer Data Byte 0 Data for a message stored in the receive FIFO buffer can be read.
23 to 16	RFDB2[7:0]	All 0	R	
15 to 8	RFDB1[7:0]	All 0	R	
7 to 0	RFDB0[7:0]	All 0	R	

When the RFDLC[3:0] value in the RSCANnRFPTRx register is smaller than 1000b, data bytes for which no data is set are read as H’00.

28.3.7.7 RSCANnRFDF1_x — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)

Access Size: RSCANnRFDF1_x register can be read only in 32-bit units

RSCANnRFDF1_xL, RSCANnRFDF1_xH registers can be read only in 16-bit units

RSCANnRFDF1_xLL, RSCANnRFDF1_xLH, RSCANnRFDF1_xHL, RSCANnRFDF1_xHH registers can be read only in 8-bit units

Address(es): RSCANnRFDF1_x: <RSCFDn_base> + H’0E0C + (H’10 × x)
RSCANnRFDF1_xL: <RSCFDn_base> + H’0E0C + (H’10 × x),
RSCANnRFDF1_xH: <RSCFDn_base> + H’0E0E + (H’10 × x)
RSCANnRFDF1_xLL: <RSCFDn_base> + H’0E0C + (H’10 × x),
RSCANnRFDF1_xLH: <RSCFDn_base> + H’0E0D + (H’10 × x),
RSCANnRFDF1_xHL: <RSCFDn_base> + H’0E0E + (H’10 × x),
RSCANnRFDF1_xHH: <RSCFDn_base> + H’0E0F + (H’10 × x)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB7[7:0]								RFDB6[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB5[7:0]								RFDB4[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RFDB7[7:0]	All 0	R	Receive FIFO Buffer Data Byte 7 Receive FIFO Buffer Data Byte 6 Receive FIFO Buffer Data Byte 5 Receive FIFO Buffer Data Byte 4 Data for a message stored in the receive FIFO buffer can be read.
23 to 16	RFDB6[7:0]	All 0	R	
15 to 8	RFDB5[7:0]	All 0	R	
7 to 0	RFDB4[7:0]	All 0	R	

When the RFDLC[3:0] value in the RSCANnRFPTRx register is smaller than 1000b, data bytes for which no data is set are read as H’00.

28.3.8 Details of Transmit/Receive FIFO Buffer-related Registers

28.3.8.1 RSCANnCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)

Access Size:	RSCANnCFCCk register can be read/written in 32-bit units RSCANnCFCCkL, RSCANnCFCCkH registers can be read/written in 16-bit units RSCANnCFCCkLL, RSCANnCFCCkLH, RSCANnCFCCkHL, RSCANnCFCCkHH registers can be read/written in 8-bit units
Address(es):	RSCANnCFCCk: <RSCFDn_base> + H'0118 + (H'04 × k) RSCANnCFCCkL: <RSCFDn_base> + H'0118 + (H'04 × k), RSCANnCFCCkH: <RSCFDn_base> + H'011A + (H'04 × k) RSCANnCFCCkLL: <RSCFDn_base> + H'0118 + (H'04 × k), RSCANnCFCCkLH: <RSCFDn_base> + H'0119 + (H'04 × k), RSCANnCFCCkHL: <RSCFDn_base> + H'011A + (H'04 × k), RSCANnCFCCkHH: <RSCFDn_base> + H'011B + (H'04 × k)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	—	—	—	—	CFTXIE	CFRXIE	CFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFITT[7:0]	All 0	R	Set a message transmission interval. Set Value: H'00 to H'FF
23 to 20	CFTML[3:0]	All 0	R	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	0	R/W	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk by (ITRCP[15:0] bits) 1: Clock dividing pclk by (ITRCP[15:0] bits × 10)
18	CFITSS	0	R/W	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the bit time clock for the channel to which the FIFO is linked.
17 to 16	CFM[1:0]	All 0	R/W	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	CFIGCV[2:0]	All 0	R/W	Transmit/Receive FIFO Receive Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	CFIM	0	R/W	Transmit/Receive FIFO Interrupt Source Select 0: ● Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. ● Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. 1: ● Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. ● Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.
11	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
10 to 8	CFDC[2:0]	All 0	R/W	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	CFTXIE	0	R/W	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
1	CFRXIE	0	R/W	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
0	CFE	0	R/W	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01b (transmit mode) or 10b (gateway mode).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01b (transmit mode) or 10b (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as m = k/3 (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be ((16 × m) + CFTML[3:0]).

See **Table 28.11** and **Table 28.12**, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p. Setting the CFDC[2:0] bits to 001b or more enables the setting of the CFTML[3:0] bits.

Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFITR Bit

This bit is enabled when the CFITSS bit is 0.

When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCANnGCFG register.

When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCANnGCFG register × 10).

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.

When this bit is 1, the bit time clock of the channel to which the FIFO is linked is the count source of the interval timer. Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFM[1:0] Bits

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to 00b (receive mode) or 10b (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).

When the CFDC[2:0] bits are set to 001b (4 messages), set the CFIGCV[2:0] bits to 001b, 011b, 101b, or 111b. Modify these bits only in global reset mode.

CFIM Bit

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFDC[2:0] Bits

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000b, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCANnCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFRXIF flag in the RSCANnCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.

Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.

When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/ receive FIFO buffer becomes empty immediately.

This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

After all other bits in the RSCANnCFCCk register have been set, set this bit to 1 by using another instruction.

28.3.8.2 RSCANnCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)

Access Size:	RSCANnCFSTSk register can be read/written in 32-bit units RSCANnCFSTSkL, RSCANnCFSTSkH registers can be read/written in 16-bit units RSCANnCFSTSkLL, RSCANnCFSTSkLH, RSCANnCFSTSkHL, RSCANnCFSTSkHH registers can be read/written in 8-bit units
Address(es):	RSCANnCFSTSk: <RSCFDn_base> + H’0178 + (H’04 × k) RSCANnCFSTSkL: <RSCFDn_base> + H’0178 + (H’04 × k), RSCANnCFSTSkH: <RSCFDn_base> + H’017A + (H’04 × k) RSCANnCFSTSkLL: <RSCFDn_base> + H’0178 + (H’04 × k), RSCANnCFSTSkLH: <RSCFDn_base> + H’0179 + (H’04 × k), RSCANnCFSTSkHL: <RSCFDn_base> + H’017A + (H’04 × k), RSCANnCFSTSkHH: <RSCFDn_base> + H’017B + (H’04 × k)
Initial Value:	H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	CFMC[7:0]	All 0	R	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
4	CFTXIF	0	R/W*1	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.
3	CFRXIF	0	R/W*1	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	0	R/W*1	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	0	R	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	1	R	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCANnCFCCk register.

- When CFM[1:0] value is 01b (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00b (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10b (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00b: In global reset mode
- When CFM[1:0] value is 01b or 10b: In channel reset mode
- When the CFE bit in the RSCANnCFCCk register is cleared to 0.

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01b or 10b, and the factor selected by the CFIM bit in the RSCANnCFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00b or 10b, and the factor selected by the CFIM bit in the RSCANnCFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

Write 0 to this flag in global operating mode or global test mode

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCANnCFCCk register.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCANnCFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00b: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00b or 10b: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01b: A value of H’FF has been written to the RSCANnCFPCTRk register after data was written to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers.

NOTE

To clear CFTXIF, CFRXIF, or CFMLT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

28.3.8.3 RSCANnCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)

Access Size:	RSCANnCFPCTRk register can only be written in 32-bit units RSCANnCFPCTRkL, RSCANnCFPCTRkH registers can only be written in 16-bit units RSCANnCFPCTRkLL, RSCANnCFPCTRkLH, RSCANnCFPCTRkHL, RSCANnCFPCTRkHH registers can only be written in 8-bit units
Address(es):	RSCANnCFPCTRk: <RSCFDn_base> + H’01D8 + (H’04 × k) RSCANnCFPCTRkL: <RSCFDn_base> + H’01D8 + (H’04 × k), RSCANnCFPCTRkH: <RSCFDn_base> + H’01DA + (H’04 × k) RSCANnCFPCTRkLL: <RSCFDn_base> + H’01D8 + (H’04 × k), RSCANnCFPCTRkLH: <RSCFDn_base> + H’01D9 + (H’04 × k), RSCANnCFPCTRkHL: <RSCFDn_base> + H’01DA + (H’04 × k), RSCANnCFPCTRkHH: <RSCFDn_base> + H’01DB + (H’04 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should be the value after reset.
7 to 0	CFPC[7:0]	All 0	W	Transmit/Receive FIFO Pointer Control ● Receive mode: Writing H’FF to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. ● Transmit mode: Writing H’FF to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. ● Gateway mode: Setting prohibited

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCANnCFCCk register is 00b):
  Writing H'FF to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCANnCFSTSk register is decremented. Read the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers to read messages from the transmit/receive FIFO buffer, and then write H'FF to the CFPC[7:0] bits.
  When writing H'FF to these bits, make sure that the CFE bit in the RSCANnCFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCANnCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCANnCFCCk register is 01b):
  Writing H'FF to the CFPC[7:0] bits stores the data written to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers before writing H'FF to the CFPC[7:0] bits.
  When writing H'FF to these bits, make sure that the CFE bit in the RSCANnCFCCk register is set to 1 and the CFFLL flag in the RSCANnCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCANnCFCCk register is 10b):
  Setting prohibited

28.3.8.4 RSCANnCFIDk — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)

Access Size:	RSCANnCFIDk register can be read/written in 32-bit units RSCANnCFIDkL, RSCANnCFIDkH registers can be read/written in 16-bit units RSCANnCFIDkLL, RSCANnCFIDkLH, RSCANnCFIDkHL, RSCANnCFIDkHH registers can be read/written in 8-bit units
Address(es):	RSCANnCFIDk: <RSCFDn_base> + H’0E80 + (H’10 × k) RSCANnCFIDkL: <RSCFDn_base> + H’0E80 + (H’10 × k), RSCANnCFIDkH: <RSCFDn_base> + H’0E82 + (H’10 × k) RSCANnCFIDkLL: <RSCFDn_base> + H’0E80 + (H’10 × k), RSCANnCFIDkLH: <RSCFDn_base> + H’0E81 + (H’10 × k), RSCANnCFIDkHL: <RSCFDn_base> + H’0E82 + (H’10 × k), RSCANnCFIDkHH: <RSCFDn_base> + H’0E83 + (H’10 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CFIDE	0	R/W	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	CFRTR	0	R/W	Transmit/Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	0	R/W	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01b (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	CFID[28:0]	All 0	R/W	Transmit/Receive FIFO Buffer ID Data • When CFM[1:0] value is 01b (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. • When CFM[1:0] value is 00b (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). This RSCFDnCFIDk register should not be read or written when the CFM[1:0] value is 10b (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b. When the CFM[1:0] value is 01b, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

This bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b. When the CFM[1:0] value is 01b, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, and buffer type) of transmit messages is stored in the transmit history buffer after transmission is completed.

This bit is enabled when the CFM[1:0] value is 01b (transmit mode).

CFID[28:0] Bits

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b.

When the CFM[1:0] value is 01b, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

28.3.8.5 RSCANnCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5)

Access Size:	RSCANnCFPTRk register can be read/written in 32-bit units RSCANnCFPTRkL, RSCANnCFPTRkH registers can be read/written in 16-bit units RSCANnCFPTRkLL, RSCANnCFPTRkLH, RSCANnCFPTRkHL, RSCANnCFPTRkHH registers can be read/written in 8-bit units
Address(es):	RSCANnCFPTRk: <RSCFDn_base> + H’0E84 + (H’10 × k) RSCANnCFPTRkL: <RSCFDn_base> + H’0E84 + (H’10 × k), RSCANnCFPTRkH: <RSCFDn_base> + H’0E86 + (H’10 × k) RSCANnCFPTRkLL: <RSCFDn_base> + H’0E84 + (H’10 × k), RSCANnCFPTRkLH: <RSCFDn_base> + H’0E85 + (H’10 × k), RSCANnCFPTRkHL: <RSCFDn_base> + H’0E86 + (H’10 × k), RSCANnCFPTRkHH: <RSCFDn_base> + H’0E87 + (H’10 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	CFDLC[3:0]	All 0	R/W	Transmit/Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	CFPTR[11:0]	All 0	R/W	Transmit/Receive FIFO Buffer Label Data ● When CFM[1:0] value is 01b (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid. ● When CFM[1:0] value is 00b (receive mode): The label information of the received message can be read.
15 to 0	CFTS[15:0]	All 0	R/W	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00b (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). This register should not be read or written when the CFM[1:0] value is 10b (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b. When the CFM[1:0] value is 01b, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer. If the data length is set to 1001b or more, the actual transmit data defaults to 8 bytes.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b. When the CFM[1:0] value is 01b, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer. These bits are valid when the CFM[1:0] value is 00b.

28.3.8.6 RSCANnCFDF0_k — Transmit/receive FIFO Buffer Access Data Field 0 Register (k = 0 to 5)

Access Size:	RSCANnCFDF0_k register can be read/written in 32-bit units RSCANnCFDF0_kL, RSCANnCFDF0_kH registers can be read/written in 16-bit units RSCANnCFDF0_kLL, RSCANnCFDF0_kLH, RSCANnCFDF0_kHL, RSCANnCFDF0_kHH registers can be read/written in 8-bit units
Address(es):	RSCANnCFDF0_k: <RSCFDn_base> + H’0E88 + (H’10 × k) RSCANnCFDF0_kL: <RSCFDn_base> + H’0E88 + (H’10 × k), RSCANnCFDF0_kH: <RSCFDn_base> + H’0E8A + (H’10 × k) RSCANnCFDF0_kLL: <RSCFDn_base> + H’0E88 + (H’10 × k), RSCANnCFDF0_kLH: <RSCFDn_base> + H’0E89 + (H’10 × k), RSCANnCFDF0_kHL: <RSCFDn_base> + H’0E8A + (H’10 × k), RSCANnCFDF0_kHH: <RSCFDn_base> + H’0E8B + (H’10 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB3[7:0]								CFDB2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB1[7:0]								CFDB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFDB3[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 3
23 to 16	CFDB2[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 2
15 to 8	CFDB1[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 1
7 to 0	CFDB0[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 0 ● When CFM[1:0] value is 01b (transmit mode): Set the transmit/receive FIFO buffer data. ● When CFM[1:0] value is 00b (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). When the CFDLC[3:0] value in the RSCANnCFPTRk register is smaller than 1000b, data bytes for which no data is set are read as H’00.

This register should not be read or written when the CFM[1:0] value is 10b (gateway mode).

28.3.8.7 RSCANnCFDF1_k — Transmit/receive FIFO Buffer Access Data Field 1 Register (k = 0 to 5)

Access Size:	RSCANnCFDF1_k register can be read/written in 32-bit units RSCANnCFDF1_kL, RSCANnCFDF1_kH registers can be read/written in 16-bit units RSCANnCFDF1_kLL, RSCANnCFDF1_kLH, RSCANnCFDF1_kHL, RSCANnCFDF1_kHH registers can be read/written in 8-bit units
Address(es):	RSCANnCFDF1_k: <RSCFDn_base> + H’0E8C + (H’10 × k) RSCANnCFDF1_kL: <RSCFDn_base> + H’0E8C + (H’10 × k), RSCANnCFDF1_kH: <RSCFDn_base> + H’0E8E + (H’10 × k) RSCANnCFDF1_kLL: <RSCFDn_base> + H’0E8C + (H’10 × k), RSCANnCFDF1_kLH: <RSCFDn_base> + H’0E8D + (H’10 × k), RSCANnCFDF1_kHL: <RSCFDn_base> + H’0E8E + (H’10 × k), RSCANnCFDF1_kHH: <RSCFDn_base> + H’0E8F + (H’10 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB7[7:0]								CFDB6[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB5[7:0]								CFDB4[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFDB7[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 7
23 to 16	CFDB6[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 6
15 to 8	CFDB5[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 5
7 to 0	CFDB4[7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 4 ● When CFM[1:0] value is 01b (transmit mode): Set the transmit/receive FIFO buffer data. ● When CFM[1:0] value is 00b (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). When the CFDLC[3:0] value in the RSCANnCFPTRk register is smaller than 1000b, data bytes for which no data is set are read as H’00.

This register should not be read or written when the CFM[1:0] value is 10b (gateway mode).

28.3.9 Details of FIFO Status-related Registers

28.3.9.1 RSCANnFESTS — FIFO Empty Status Register

Access Size:	RSCANnFESTS register can be read only in 32-bit units RSCANnFESTSL, RSCANnFESTSH registers can be read only in 16-bit units RSCANnFESTSLL, RSCANnFESTSLH, RSCANnFESTSHL, RSCANnFESTSHH registers can be read only in 8-bit units
Address(es):	RSCANnFESTS: <RSCFDn_base> + H’0238 RSCANnFESTSL: <RSCFDn_base> + H’0238, RSCANnFESTSH: <RSCFDn_base> + H’023A RSCANnFESTSLL: <RSCFDn_base> + H’0238, RSCANnFESTSLH: <RSCFDn_base> + H’0239, RSCANnFESTSHL: <RSCFDn_base> + H’023A, RSCANnFESTSHH: <RSCFDn_base> + H’023B
Initial Value:	H’03FF FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5EMP	CF4EMP	CF3EMP	CF2EMP	CF1EMP	CF0EMP	RF7EMP	RF6EMP	RF5EMP	RF4EMP	RF3EMP	RF2EMP	RF1EMP	RF0EMP
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	H’0_0FFF	R	Reserved These bits are read as the value after reset.
13	CF5EMP	1	R	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 5)
12	CF4EMP	1	R	
11	CF3EMP	1	R	
10	CF2EMP	1	R	
9	CF1EMP	1	R	
8	CF0EMP	1	R	
7	RF7EMP	1	R	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains a unread messages. 1: Receive FIFO buffer x contains no unread message. (x = 0 to 7)
6	RF6EMP	1	R	
5	RF5EMP	1	R	
4	RF4EMP	1	R	
3	RF3EMP	1	R	
2	RF2EMP	1	R	
1	RF1EMP	1	R	
0	RF0EMP	1	R	

The RSCANnFESTS register is set to H’03FF FFFF in global reset mode.

CFkEMP Flag (k = 0 to 5)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCANnCFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCANnRFSTSx register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

28.3.9.2 RSCANnFFSTS — FIFO Full Status Register

Access Size:	RSCANnFFSTS register can be read only in 32-bit units RSCANnFFSTSL, RSCANnFFSTSH registers can be read only in 16-bit units RSCANnFFSTSLL, RSCANnFFSTSLH, RSCANnFFSTSHL, RSCANnFFSTSHH registers can be read only in 8-bit units
Address(es):	RSCANnFFSTS: <RSCFDn_base> + H’023C RSCANnFFSTSL: <RSCFDn_base> + H’023C, RSCANnFFSTSH: <RSCFDn_base> + H’023E RSCANnFFSTSLL: <RSCFDn_base> + H’023C, RSCANnFFSTSLH: <RSCFDn_base> + H’023D, RSCANnFFSTSHL: <RSCFDn_base> + H’023E, RSCANnFFSTSHH: <RSCFDn_base> + H’023F
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5FLL	CF4FLL	CF3FLL	CF2FLL	CF1FLL	CF0FLL	RF7FLL	RF6FLL	RF5FLL	RF4FLL	RF3FLL	RF2FLL	RF1FLL	RF0FLL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset.
13	CF5FLL	0	R	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 0 to 5)
12	CF4FLL	0	R	
11	CF3FLL	0	R	
10	CF2FLL	0	R	
9	CF1FLL	0	R	
8	CF0FLL	0	R	
7	RF7FLL	0	R	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7)
6	RF6FLL	0	R	
5	RF5FLL	0	R	
4	RF4FLL	0	R	
3	RF3FLL	0	R	
2	RF2FLL	0	R	
1	RF1FLL	0	R	
0	RF0FLL	0	R	

The RSCANnFFSTS register is cleared to H’0000 0000 in global reset mode.

CFkFLL Flag (k = 0 to 5)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCANnCFSTSk register is set to 1 (the transmit/receive FIFO buffer is full).

When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCANnRFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

28.3.9.3 RSCANnFMSTS — FIFO Message Lost Status Register

Access Size:	RSCANnFMSTS register can be read only in 32-bit units RSCANnFMSTSL, RSCANnFMSTSH registers can be read only in 16-bit units RSCANnFMSTSLL, RSCANnFMSTSLH, RSCANnFMSTSHL, RSCANnFMSTSHH registers can be read only in 8-bit units
Address(es):	RSCANnFMSTS: <RSCFDn_base> + H’0240 RSCANnFMSTSL: <RSCFDn_base> + H’0240, RSCANnFMSTSH: <RSCFDn_base> + H’0242 RSCANnFMSTSLL: <RSCFDn_base> + H’0240, RSCANnFMSTSLH: <RSCFDn_base> + H’0241, RSCANnFMSTSHL: <RSCFDn_base> + H’0242, RSCANnFMSTSHH: <RSCFDn_base> + H’0243
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5MLT	CF4MLT	CF3MLT	CF2MLT	CF1MLT	CF0MLT	RF7MLT	RF6MLT	RF5MLT	RF4MLT	RF3MLT	RF2MLT	RF1MLT	RF0MLT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset.
13	CF5MLT	0	R	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 5)
12	CF4MLT	0	R	
11	CF3MLT	0	R	
10	CF2MLT	0	R	
9	CF1MLT	0	R	
8	CF0MLT	0	R	
7	RF7MLT	0	R	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	0	R	
5	RF5MLT	0	R	
4	RF4MLT	0	R	
3	RF3MLT	0	R	
2	RF2MLT	0	R	
1	RF1MLT	0	R	
0	RF0MLT	0	R	

The RSCANnFMSTS register is cleared to H’0000 0000 in global reset mode.

CFkMLT Flag (k = 0 to 5)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCANnRFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

28.3.9.4 RSCANnRFISTS — Receive FIFO Buffer Interrupt Flag Status Register

Access Size:	RSCANnRFISTS register can be read only in 32-bit units RSCANnRFISTSL, RSCANnRFISTSH registers can be read only in 16-bit units RSCANnRFISTSLL, RSCANnRFISTSLH, RSCANnRFISTSHL, RSCANnRFISTSHH registers can be read only in 8-bit units
Address(es):	RSCANnRFISTS: <RSCFDn_base> + H’0244 RSCANnRFISTSL: <RSCFDn_base> + H’0244, RSCANnRFISTSH: <RSCFDn_base> + H’0246 RSCANnRFISTSLL: <RSCFDn_base> + H’0244, RSCANnRFISTSLH: <RSCFDn_base> + H’0245, RSCANnRFISTSHL: <RSCFDn_base> + H’0246, RSCANnRFISTSHH: <RSCFDn_base> + H’0247
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as the value after reset.
7	RF7IF	0	R	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	0	R	
5	RF5IF	0	R	
4	RF4IF	0	R	
3	RF3IF	0	R	
2	RF2IF	0	R	
1	RF1IF	0	R	
0	RF0IF	0	R	

The RSCANnRFISTS register is cleared to H’0000 0000 in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCANnRFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

28.3.9.5 RSCANnCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register

Access Size:	RSCANnCFRISTS register can be read only in 32-bit units RSCANnCFRISTSL, RSCANnCFRISTSH registers can be read only in 16-bit units RSCANnCFRISTSLL, RSCANnCFRISTSLH, RSCANnCFRISTSHL, RSCANnCFRISTSHH registers can be read only in 8-bit units
Address(es):	RSCANnCFRISTS: <RSCFDn_base> + H’0248 RSCANnCFRISTSL: <RSCFDn_base> + H’0248, RSCANnCFRISTSH: <RSCFDn_base> + H’024A RSCANnCFRISTSLL: <RSCFDn_base> + H’0248, RSCANnCFRISTSLH: <RSCFDn_base> + H’0249, RSCANnCFRISTSHL: <RSCFDn_base> + H’024A, RSCANnCFRISTSHH: <RSCFDn_base> + H’024B
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5RXIF	CF4RXIF	CF3RXIF	CF2RXIF	CF1RXIF	CF0RXIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are read as the value after reset.
5	CF5RXIF	0	R	Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k receive interrupt request is present. 1: A transmit/receive FIFO buffer k receive interrupt request is present. (k = 0 to 5)
4	CF4RXIF	0	R	
3	CF3RXIF	0	R	
2	CF2RXIF	0	R	
1	CF1RXIF	0	R	
0	CF0RXIF	0	R	

The RSCANnCFRISTS register is cleared to H’0000 0000 in global reset mode.

CFkRXIF Flag (k = 0 to 5)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

28.3.9.6 RSCANnCFTISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register

Access Size:	RSCANnCFTISTS register can be read only in 32-bit units RSCANnCFTISTSL, RSCANnCFTISTSH registers can be read only in 16-bit units RSCANnCFTISTSLL, RSCANnCFTISTSLH, RSCANnCFTISTSHL, RSCANnCFTISTSHH registers can be read only in 8-bit units
Address(es):	RSCANnCFTISTS: <RSCFDn_base> + H’024C RSCANnCFTISTSL: <RSCFDn_base> + H’024C, RSCANnCFTISTSH: <RSCFDn_base> + H’024E RSCANnCFTISTSLL: <RSCFDn_base> + H’024C, RSCANnCFTISTSLH: <RSCFDn_base> + H’024D, RSCANnCFTISTSHL: <RSCFDn_base> + H’024E, RSCANnCFTISTSHH: <RSCFDn_base> + H’024F
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5TXIF	CF4TXIF	CF3TXIF	CF2TXIF	CF1TXIF	CF0TXIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are read as the value after reset.
5	CF5TXIF	0	R	Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. (k = 0 to 5)
4	CF4TXIF	0	R	
3	CF3TXIF	0	R	
2	CF2TXIF	0	R	
1	CF1TXIF	0	R	
0	CF0TXIF	0	R	

The RSCANnCFTISTS register is cleared to H’0000 0000 in global reset mode.

CFkTXIF Flag (k = 0 to 5)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

28.3.10 Details of Transmit Buffer-related Registers

28.3.10.1 RSCANnTMCp — Transmit Buffer Control Register (p = 0 to 31)

Access Size:	RSCANnTMCp registers can be read/written in 8-bit units
Address(es):	RSCANnTMCp: <RSCFDn_base> + H'0250 + (H'01 × p)
Initial Value:	H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	TMOM	0	R/W	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	0	R/W*1	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	0	R/W*1	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

When the RSCANnTMCp register meets any of the following conditions, set it to H'00.

- The RSCANnTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCANnCFCCk register (p = m × 16 + the value of CFTML[3:0] bits).
- The RSCANnTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCANnTXQCCm (m = 0, 1) register (p = (m × 16 + 15) to (m × 16 + 15 – the value of TXQDC[3:0] bits)).

Bits in the RSCANnTMCp register are all cleared to 0 in channel reset mode. Modify the RSCANnTMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCANnTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

TMTAR Bit

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.

The TMTAR bit can be set to 1 when TMTR bit is 1.

The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.

The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCANnTMSTSp register is 00b.

28.3.10.2 RSCANnTMSTSp — Transmit Buffer Status Register (p = 0 to 31)

Access Size: RSCANnTMSTSp registers can be read/written in 8-bit units

Address(es): RSCANnTMSTSp: <RSCFDn_base> + H'02D0 + (H'01 × p)

Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
4	TMTARM	0	R	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	0	R	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	All 0	R/W	Transmit Buffer Transmit Result Status Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	0	R	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCANnTMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCANnTMCp register is set to 1.

The TMTARM flag is set to 0 when the TMTAR bit in the RSCANnTMCp register is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCANnTMCp register is set to 1.

The TMTRM flag is set to 0 when the TMTR bit in the RSCANnTMCp register is set to 0.

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.

00b: Transmission is in progress or no transmit request is present.

01b: Transmission from the transmit buffer was aborted.

10b: Transmission has been completed with the TMTAR bit in the RSCANnTMCp register set to 0 (transmit abort is not requested).

11b: Transmission has been completed with the TMTAR bit in the RSCANnTMCp register set to 1 (transmit abort is requested).

Write 00b to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00b to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

28.3.10.3 RSCANnTMIDp — Transmit Buffer ID Register (p = 0 to 31)

Access Size:	RSCANnTMIDp register can be read/written in 32-bit units RSCANnTMIDpL, RSCANnTMIDpH registers can be read/written in 16-bit units RSCANnTMIDpLL, RSCANnTMIDpLH, RSCANnTMIDpHL, RSCANnTMIDpHH registers can be read/written in 8-bit units
Address(es):	RSCANnTMIDp: <RSCFDn_base> + H’1000 + (H’10 × p) RSCANnTMIDpL: <RSCFDn_base> + H’1000 + (H’10 × p), RSCANnTMIDpH: <RSCFDn_base> + H’1002 + (H’10 × p) RSCANnTMIDpLL: <RSCFDn_base> + H’1000 + (H’10 × p), RSCANnTMIDpLH: <RSCFDn_base> + H’1001 + (H’10 × p), RSCANnTMIDpHL: <RSCFDn_base> + H’1002 + (H’10 × p), RSCANnTMIDpHH: <RSCFDn_base> + H’1003 + (H’10 × p)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TMIDE	0	R/W	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	0	R/W	Transmit Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	0	R/W	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	All 0	R/W	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp (timestamp is included if the TMTSCE bit in the RSCANnGCFG register is 1)) of transmit messages is stored in the transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

28.3.10.4 RSCANnTMPTRp — Transmit Buffer Pointer Register (p = 0 to 31)

Access Size:	RSCANnTMPTRp register can be read/written in 32-bit units RSCANnTMPTRpL, RSCANnTMPTRpH registers can be read/written in 16-bit units RSCANnTMPTRpLL, RSCANnTMPTRpLH, RSCANnTMPTRpHL, RSCANnTMPTRpHH registers can be read/written in 8-bit units
Address(es):	RSCANnTMPTRp: <RSCFDn_base> + H'1004 + (H'10 × p) RSCANnTMPTRpL: <RSCFDn_base> + H'1004 + (H'10 × p), RSCANnTMPTRpH: <RSCFDn_base> + H'1006 + (H'10 × p) RSCANnTMPTRpLL: <RSCFDn_base> + H'1004 + (H'10 × p), RSCANnTMPTRpLH: <RSCFDn_base> + H'1005 + (H'10 × p), RSCANnTMPTRpHL: <RSCFDn_base> + H'1006 + (H'10 × p), RSCANnTMPTRpHH: <RSCFDn_base> + H'1007 + (H'10 × p)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	TMDLC[3:0]	All 0	R/W	Transmit Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 24	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
23 to 16	TMPTR[7:0]	All 0	R/W	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCANnTMIDp register is set to 0 (data frame). If the data length is set to 1001b or more, the transmit data is 8 bytes long.

When the TMRTR bit is set to 1 (remote frame), set the data length of messages to be requested.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

28.3.10.5 RSCANnTMDF0_p — Transmit Buffer Data Field 0 Register (p = 0 to 31)

Access Size: RSCANnTMDF0_p register can be read/written in 32-bit units
RSCANnTMDF0_pL, RSCANnTMDF0_pH registers can be read/written in 16-bit units
RSCANnTMDF0_pLL, RSCANnTMDF0_pLH, RSCANnTMDF0_pHL, RSCANnTMDF0_pHH registers can be read/written in 8-bit units

Address(es): RSCANnTMDF0_p: <RSCFDn_base> + H’1008 + (H’10 × p)
RSCANnTMDF0_pL: <RSCFDn_base> + H’1008 + (H’10 × p),
RSCANnTMDF0_pH: <RSCFDn_base> + H’100A + (H’10 × p)
RSCANnTMDF0_pLL: <RSCFDn_base> + H’1008 + (H’10 × p),
RSCANnTMDF0_pLH: <RSCFDn_base> + H’1009 + (H’10 × p),
RSCANnTMDF0_pHL: <RSCFDn_base> + H’100A + (H’10 × p),
RSCANnTMDF0_pHH: <RSCFDn_base> + H’100B + (H’10 × p)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB3[7:0]								TMDB2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB1[7:0]								TMDB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TMDB3[7:0]	All 0	R/W	Transmit Buffer Data Byte 3
23 to 16	TMDB2[7:0]	All 0	R/W	Transmit Buffer Data Byte 2
15 to 8	TMDB1[7:0]	All 0	R/W	Transmit Buffer Data Byte 1
7 to 0	TMDB0[7:0]	All 0	R/W	Transmit Buffer Data Byte 0 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

28.3.10.6 RSCANnTMDF1_p — Transmit Buffer Data Field 1 Register (p = 0 to 31)

Access Size:	RSCANnTMDF1_p register can be read/written in 32-bit units RSCANnTMDF1_pL, RSCANnTMDF1_pH registers can be read/written in 16-bit units RSCANnTMDF1_pLL, RSCANnTMDF1_pLH, RSCANnTMDF1_pHL, RSCANnTMDF1_pHH registers can be read/written in 8-bit units
Address(es):	RSCANnTMDF1_p: <RSCFDn_base> + H’100C + (H’10 × p) RSCANnTMDF1_pL: <RSCFDn_base> + H’100C + (H’10 × p), RSCANnTMDF1_pH: <RSCFDn_base> + H’100E + (H’10 × p) RSCANnTMDF1_pLL: <RSCFDn_base> + H’100C + (H’10 × p), RSCANnTMDF1_pLH: <RSCFDn_base> + H’100D + (H’10 × p), RSCANnTMDF1_pHL: <RSCFDn_base> + H’100E + (H’10 × p), RSCANnTMDF1_pHH: <RSCFDn_base> + H’100F + (H’10 × p)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB7[7:0]								TMDB6[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB5[7:0]								TMDB4[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TMDB7[7:0]	All 0	R/W	Transmit Buffer Data Byte 7
23 to 16	TMDB6[7:0]	All 0	R/W	Transmit Buffer Data Byte 6
15 to 8	TMDB5[7:0]	All 0	R/W	Transmit Buffer Data Byte 5
7 to 0	TMDB4[7:0]	All 0	R/W	Transmit Buffer Data Byte 4 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

28.3.10.7 RSCANnTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0)

Access Size: RSCANnTMIECy register can be read/written in 32-bit units
RSCANnTMIECyL, RSCANnTMIECyH registers can be read/written in 16-bit units
RSCANnTMIECyLL, RSCANnTMIECyLH, RSCANnTMIECyHL, RSCANnTMIECyHH registers can be read/written in 8-bit units

Address(es): RSCANnTMIECy: <RSCFDn_base> + H’0390 + (H’04 × y)
RSCANnTMIECyL: <RSCFDn_base> + H’0390 + (H’04 × y),
RSCANnTMIECyH: <RSCFDn_base> + H’0392 + (H’04 × y)
RSCANnTMIECyLL: <RSCFDn_base> + H’0390 + (H’04 × y),
RSCANnTMIECyLH: <RSCFDn_base> + H’0391 + (H’04 × y),
RSCANnTMIECyHL: <RSCFDn_base> + H’0392 + (H’04 × y),
RSCANnTMIECyHH: <RSCFDn_base> + H’0393 + (H’04 × y)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMIEp	All 0	R/W	Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled
15 to 0	TMIEp	All 0	R/W	Transmit Buffer Interrupt Enable p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 31)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCANnTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

Table 28.15 shows the bit assignment.

Table 28.15 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.3.11 Details of Transmit Buffer Status-related Registers

28.3.11.1 RSCANnTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)

Access Size:	RSCANnTMTRSTSy register can be read only in 32-bit units RSCANnTMTRSTSyL, RSCANnTMTRSTSyH registers can be read only in 16-bit units RSCANnTMTRSTSyLL, RSCANnTMTRSTSyLH, RSCANnTMTRSTSyHL, RSCANnTMTRSTSyHH registers can be read only in 8-bit units
Address(es):	RSCANnTMTRSTSy: <RSCFDn_base> + H’0350 + (H’04 × y) RSCANnTMTRSTSyL: <RSCFDn_base> + H’0350 + (H’04 × y), RSCANnTMTRSTSyH: <RSCFDn_base> + H’0352 + (H’04 × y) RSCANnTMTRSTSyLL: <RSCFDn_base> + H’0350 + (H’04 × y), RSCANnTMTRSTSyLH: <RSCFDn_base> + H’0351 + (H’04 × y), RSCANnTMTRSTSyHL: <RSCFDn_base> + H’0352 + (H’04 × y), RSCANnTMTRSTSyHH: <RSCFDn_base> + H’0353 + (H’04 × y)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTRSTSp	All 0	R	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit request is present. 1: A transmit request is present.
15 to 0	TMTRSTSp	All 0	R	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTR bit in the RSCANnTMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 28.16 shows the bit assignment.

Table 28.16 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.3.11.2 RSCANnTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)

Access Size:	RSCANnTMTARSTSy register can be read only in 32-bit units RSCANnTMTARSTSyL, RSCANnTMTARSTSyH registers can be read only in 16-bit units RSCANnTMTARSTSyLL, RSCANnTMTARSTSyLH, RSCANnTMTARSTSyHL, RSCANnTMTARSTSyHH registers can be read only in 8- bit units
Address(es):	RSCANnTMTARSTSy: <RSCFDn_base> + H’0360 + (H’04 × y) RSCANnTMTARSTSyL: <RSCFDn_base> + H’0360 + (H’04 × y), RSCANnTMTARSTSyH: <RSCFDn_base> + H’0362 + (H’04 × y) RSCANnTMTARSTSyLL: <RSCFDn_base> + H’0360 + (H’04 × y), RSCANnTMTARSTSyLH: <RSCFDn_base> + H’0361 + (H’04 × y), RSCANnTMTARSTSyHL: <RSCFDn_base> + H’0362 + (H’04 × y), RSCANnTMTARSTSyHH: <RSCFDn_base> + H’0363 + (H’04 × y)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTARSTSp	All 0	R	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit abort request is present. 1: A transmit abort request is present.
15 to 0	TMTARSTSp	All 0	R	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

TMTARSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTAR bit in the RSCANnTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 28.17 shows the bit assignment.

Table 28.17 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.3.11.3 RSCANnTMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)

Access Size:	RSCFDnTMTCSTSyRSCANnTMTCSTSy register can be read only in 32-bit units RSCANnTMTCSTSyL, RSCANnTMTCSTSyH registers can be read only in 16-bit units RSCANnTMTCSTSyLL, RSCANnTMTCSTSyLH, RSCANnTMTCSTSyHL, RSCANnTMTCSTSyHH registers can be read only in 8-bit units
Address(es):	RSCANnTMTCSTSy: <RSCFDn_base> + H’0370 + (H’04 × y) RSCANnTMTCSTSyL: <RSCFDn_base> + H’0370 + (H’04 × y), RSCANnTMTCSTSyH: <RSCFDn_base> + H’0372 + (H’04 × y) RSCANnTMTCSTSyLL: <RSCFDn_base> + H’0370 + (H’04 × y), RSCANnTMTCSTSyLH: <RSCFDn_base> + H’0371 + (H’04 × y), RSCANnTMTCSTSyHL: <RSCFDn_base> + H’0372 + (H’04 × y), RSCANnTMTCSTSyHH: <RSCFDn_base> + H’0373 + (H’04 × y)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTCSTSp	All 0	R	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission has not been completed. 1: Transmission has been completed.
15 to 0	TMTCSTSp	All 0	R	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCANnTMSTSp register is set to 10b (transmission has been completed (without transmit abort request)) or 11b (transmission has been completed (with transmit abort request)), the corresponding TMTCSTSp flag is set to 1.

A TMTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00b or in channel reset mode.

Table 28.18 shows the bit assignment.

Table 28.18 TMTCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.3.11.4 RSCANnTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0)

Access Size:	RSCANnTMTASTSy register can be read only in 32-bit units RSCANnTMTASTSyL, RSCANnTMTASTSyH registers can be read only in 16-bit units RSCANnTMTASTSyLL, RSCANnTMTASTSyLH, RSCANnTMTASTSyHL, RSCANnTMTASTSyHH registers can be read only in 8-bit units
Address(es):	RSCANnTMTASTSy: <RSCFDn_base> + H'0380 + (H'04 × y) RSCANnTMTASTSyL: <RSCFDn_base> + H'0380 + (H'04 × y), RSCANnTMTASTSyH: <RSCFDn_base> + H'0382 + (H'04 × y) RSCANnTMTASTSyLL: <RSCFDn_base> + H'0380 + (H'04 × y), RSCANnTMTASTSyLH: <RSCFDn_base> + H'0381 + (H'04 × y), RSCANnTMTASTSyHL: <RSCFDn_base> + H'0382 + (H'04 × y), RSCANnTMTASTSyHH: <RSCFDn_base> + H'0383 + (H'04 × y)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31 to y x 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTASTSp	All 0	R	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted 1: Transmission is aborted
15 to 0	TMTASTSp	All 0	R	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCANnTMSTSp register is set to 01b (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00b or in channel reset mode.

Table 28.19 shows the bit assignment.

Table 28.19 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.3.12 Details of Transmit Queue-related Registers

28.3.12.1 RSCANnTXQCCm — Transmit Queue Configuration and Control Register (m = 0, 1)

Access Size:	RSCANnTXQCCm register can be read/written in 32-bit units RSCANnTXQCCmL, RSCANnTXQCCmH registers can be read/written in 16-bit units RSCANnTXQCCmLL, RSCANnTXQCCmLH, RSCANnTXQCCmHL, RSCANnTXQCCmHH registers can be read/written in 8-bit units
Address(es):	RSCANnTXQCCm: <RSCFDn_base> + H’03A0 + (H’04 × m) RSCANnTXQCCmL: <RSCFDn_base> + H’03A0 + (H’04 × m), RSCANnTXQCCmH: <RSCFDn_base> + H’03A2 + (H’04 × m) RSCANnTXQCCmLL: <RSCFDn_base> + H’03A0 + (H’04 × m), RSCANnTXQCCmLH: <RSCFDn_base> + H’03A1 + (H’04 × m), RSCANnTXQCCmHL: <RSCFDn_base> + H’03A2 + (H’04 × m), RSCANnTXQCCmHH: <RSCFDn_base> + H’03A3 + (H’04 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
13	TXQIM	0	R/W	Transmit Queue Interrupt Source Select 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	0	R/W	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	All 0	R/W	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited.
7 to 1	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
0	TXQE	0	R/W	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated. Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see **Figure 28.9**. Modify these bits only in channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to 0010b or more.

28.3.12.2 RSCANnTXQSTSm — Transmit Queue Status Register (m = 0, 1)

Access Size:	RSCANnTXQSTSm register can be read/written in 32-bit units RSCANnTXQSTSmL, RSCANnTXQSTSmH registers can be read/written in 16-bit units RSCANnTXQSTSmLL, RSCANnTXQSTSmLH, RSCANnTXQSTSmHL, RSCANnTXQSTSmHH registers can be read/written in 8-bit units
Address(es):	RSCANnTXQSTSm: <RSCFDn_base> + H’03C0 + (H’04 × m) RSCANnTXQSTSmL: <RSCFDn_base> + H’03C0 + (H’04 × m), RSCANnTXQSTSmH: <RSCFDn_base> + H’03C2 + (H’04 × m) RSCANnTXQSTSmLL: <RSCFDn_base> + H’03C0 + (H’04 × m), RSCANnTXQSTSmLH: <RSCFDn_base> + H’03C1 + (H’04 × m), RSCANnTXQSTSmHL: <RSCFDn_base> + H’03C2 + (H’04 × m), RSCANnTXQSTSmHH: <RSCFDn_base> + H’03C3 + (H’04 × m)
Initial Value:	H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
12 to 8	—	All 0	R	Reserved When read, an undefined value is returned.The write value should be the value after reset.
7 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	TXQIF	0	R/W*1	Transmit Queue Interrupt Request Flag 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.
1	TXQFLL	0	R	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	1	R	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCANnTXQCCm register has occurred. The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCANnTXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCANnTXQCCm register.

This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.

This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

28.3.12.3 RSCANnTXQPCTRm — Transmit Queue Pointer Control Register (m = 0, 1)

Access Size: RSCANnTXQPCTRm register can only be written in 32-bit units

RSCANnTXQPCTRmL, RSCANnTXQPCTRmH registers can only be written in 16-bit units

RSCANnTXQPCTRmLL, RSCANnTXQPCTRmLH, RSCANnTXQPCTRmHL, RSCANnTXQPCTRmHH registers can only be written in 8-bit units

Address(es): RSCANnTXQPCTRm: <RSCFDn_base> + H’03E0 + (H’04 × m)
RSCANnTXQPCTRmL: <RSCFDn_base> + H’03E0 + (H’04 × m),
RSCANnTXQPCTRmH: <RSCFDn_base> + H’03E2 + (H’04 × m)
RSCANnTXQPCTRmLL: <RSCFDn_base> + H’03E0 + (H’04 × m),
RSCANnTXQPCTRmLH: <RSCFDn_base> + H’03E1 + (H’04 × m),
RSCANnTXQPCTRmHL: <RSCFDn_base> + H’03E2 + (H’04 × m),
RSCANnTXQPCTRmHH: <RSCFDn_base> + H’03E3 + (H’04 × m)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should be the value after reset.
7 to 0	TXQPC[7:0]	All 0	W	Transmit Queue Pointer Control Writing H’FF to these bits moves the write pointer of the transmit queue to the next queue buffer.

TXQPC[7:0] Bits

Writing H’FF to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCANnTMIDp, RSCANnTMPTRp, RSCANnTMDF0_p, and RSCANnTMDF1_p registers (p = 15, 31) before writing H’FF to the TXQPC[7:0] bits.

When writing H’FF to these bits, make sure that the TXQE bit in the RSCANnTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCANnTXQSTSm register is 0 (the transmit queue is not full).

28.3.13 Details of Transmit history-related Registers

28.3.13.1 RSCANnTHLCCm — Transmit History Configuration and Control Register (m = 0, 1)

Access Size: RSCANnTHLCCm register can be read/written in 32-bit units
RSCANnTHLCCmL, RSCANnTHLCCmH registers can be read/written in 16-bit units
RSCANnTHLCCmLL, RSCANnTHLCCmLH, RSCANnTHLCCmHL, RSCANnTHLCCmHH registers can be read/written in 8-bit units

Address(es): RSCANnTHLCCm: <RSCFDn_base> + H’0400 + (H’04 × m)
RSCANnTHLCCmL: <RSCFDn_base> + H’0400 + (H’04 × m),
RSCANnTHLCCmH: <RSCFDn_base> + H’0402 + (H’04 × m)
RSCANnTHLCCmLL: <RSCFDn_base> + H’0400 + (H’04 × m),
RSCANnTHLCCmLH: <RSCFDn_base> + H’0401 + (H’04 × m),
RSCANnTHLCCmHL: <RSCFDn_base> + H’0402 + (H’04 × m),
RSCANnTHLCCmHH: <RSCFDn_base> + H’0403 + (H’04 × m)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
10	THLDTE	0	R/W	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	0	R/W	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	0	R/W	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
0	THLE	0	R/W	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.

Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer. Modify this bit in channel communication mode or channel halt mode.

This bit is cleared to 0 in channel reset mode.

28.3.13.2 RSCANnTHLSTSm — Transmit History Status Register (m = 0, 1)

Access Size:	RSCANnTHLSTSm register can be read/written in 32-bit units RSCANnTHLSTSmL, RSCANnTHLSTSmH registers can be read/written in 16-bit units RSCANnTHLSTSmLL, RSCANnTHLSTSmLH, RSCANnTHLSTSmHL, RSCANnTHLSTSmHH registers can be read/written in 8-bit units
Address(es):	RSCANnTHLSTSm: <RSCFDn_base> + H'0420 + (H'04 × m) RSCANnTHLSTSmL: <RSCFDn_base> + H'0420 + (H'04 × m), RSCANnTHLSTSmH: <RSCFDn_base> + H'0422 + (H'04 × m) RSCANnTHLSTSmLL: <RSCFDn_base> + H'0420 + (H'04 × m), RSCANnTHLSTSmLH: <RSCFDn_base> + H'0421 + (H'04 × m), RSCANnTHLSTSmHL: <RSCFDn_base> + H'0422 + (H'04 × m), RSCANnTHLSTSmHH: <RSCFDn_base> + H'0423 + (H'04 × m)
Initial Value:	H'0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THLELT	THLFLL	THLEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
12 to 8	THLMC[4:0]	All 0	R	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
3	THLIF	0	R/W*1	Transmit History Interrupt Request Flag 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present.
2	THLELT	0	R/W*1	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	0	R	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	1	R	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer.

These bits are cleared to 0 in channel reset mode.

THLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCANnTHLCCm register occurs.

This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.

When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLELT Flag

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.

When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCANnTHLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.

This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCANnTHLCCm register is set to 0 (transmit history buffer is not used).

NOTE

To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

28.3.13.3 RSCANnTHLPCTRm — Transmit History Pointer Control Register (m = 0, 1)

Access Size:	RSCANnTHLPCTRm register can only be written in 32-bit units RSCANnTHLPCTRmL, RSCANnTHLPCTRmH registers can only be written in 16-bit units RSCANnTHLPCTRmLL, RSCANnTHLPCTRmLH, RSCANnTHLPCTRmHL, RSCANnTHLPCTRmHH registers can only be written in 8- bit units
Address(es):	RSCANnTHLPCTRm: <RSCFDn_base> + H’0440 + (H’04 × m) RSCANnTHLPCTRmL: <RSCFDn_base> + H’0440 + (H’04 × m), RSCANnTHLPCTRmH: <RSCFDn_base> + H’0442 + (H’04 × m) RSCANnTHLPCTRmLL: <RSCFDn_base> + H’0440 + (H’04 × m), RSCANnTHLPCTRmLH: <RSCFDn_base> + H’0441 + (H’04 × m), RSCANnTHLPCTRmHL: <RSCFDn_base> + H’0442 + (H’04 × m), RSCANnTHLPCTRmHH: <RSCFDn_base> + H’0443 + (H’04 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved When writing to these bits, write the value after reset.
7 to 0	THLPC[7:0]	All 0	W	Transmit History List Pointer Control Writing H’FF to these bits moves the read pointer to the next unread data in the transmit history buffer.

THLPC[7:0] Bits

When the THLPC[7:0] bits are set to H’FF, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCANnTHLSTSm register is decremented. Write H’FF to the THLPC[7:0] bits after reading from the RSCANnTHLACCm register.

When writing H’FF to these bits, make sure that the THLE bit in the RSCANnTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCANnTHLSTSm register is 0.

28.3.13.4 RSCANnTHLACCm — Transmit History Access Register (m = 0, 1)

Access Size:	RSCANnTHLACCm register can be read only in 32-bit units RSCANnTHLACCmL, RSCANnTHLACCmH registers can be read only in 16-bit units RSCANnTHLACCmLL, RSCANnTHLACCmLH, RSCANnTHLACCmHL, RSCANnTHLACCmHH registers can be read only in 8-bit units
Address(es):	RSCANnTHLACCm: <RSCFDn_base> + H'1800 + (H'04 × m) RSCANnTHLACCmL: <RSCFDn_base> + H'1800 + (H'04 × m), RSCANnTHLACCmH: <RSCFDn_base> + H'1802 + (H'04 × m) RSCANnTHLACCmLL: <RSCFDn_base> + H'1800 + (H'04 × m), RSCANnTHLACCmLH: <RSCFDn_base> + H'1801 + (H'04 × m), RSCANnTHLACCmHL: <RSCFDn_base> + H'1802 + (H'04 × m), RSCANnTHLACCmHH: <RSCFDn_base> + H'1803 + (H'04 × m)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTS[15:0]	All 0	R	Timestamp Data The timestamp data of stored data can be read.
15 to 8	TID[7:0]	All 0	R	Label Data The label information of stored data can be read.
7	—	0	R	Reserved This bit is read as the value after reset.
6 to 3	BN[3:0]	All 0	R	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	All 0	R	Buffer Type Data b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 0 0 0: Transmit queue

TMTS[15:0] Bits

When the TMTSCE bit in the RSCANnGCFG register is 1, timestamp values in transmit history data stored in the transmit history buffer are displayed. When the TMTSCE bit is 0, these bits are always read as 0.

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer.

28.3.14 Details of Test-related Registers

28.3.14.1 RSCANnGTSTCFG — Global Test Configuration Register

Access Size:	RSCANnGTSTCFG register can be read/written in 32-bit units RSCANnGTSTCFGL, RSCANnGTSTCFGH registers can be read/written in 16-bit units RSCANnGTSTCFGLL, RSCANnGTSTCFGLH, RSCANnGTSTCFGHL, RSCANnGTSTCFGHH registers can be read/written in 8-bit units
Address(es):	RSCANnGTSTCFG: <RSCFDn_base> + H’0468 RSCANnGTSTCFGL: <RSCFDn_base> + H’0468, RSCANnGTSTCFGH: <RSCFDn_base> + H’046A RSCANnGTSTCFGLL: <RSCFDn_base> + H’0468, RSCANnGTSTCFGLH: <RSCFDn_base> + H’0469, RSCANnGTSTCFGHL: <RSCFDn_base> + H’046A, RSCANnGTSTCFGHH: <RSCFDn_base> + H’046B
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	C1ICBCE	C0ICBCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
22 to 16	RTMPS[6:0]	All 0	R/W	RAM Test Page Configuration Set a value within a range of page 0 (H’00) to page 19 (H’13).
15 to 2	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
1	C1ICBCE	0	R/W	CAN1 Inter-Channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	0	R/W	CAN0 Inter-Channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCANnGTSTCFG register only in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of H’00 to H’13, inclusive.

C1ICBCE Bit

Setting this bit to 1 enables the channel 1 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C0ICBCE Bit

Setting this bit to 1 enables the channel 0 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

28.3.14.2 RSCANnGTSTCTR — Global Test Control Register

Access Size:	RSCANnGTSTCTR register can be read/written in 32-bit units RSCANnGTSTCTRL, RSCANnGTSTCTRH registers can be read/written in 16-bit units RSCANnGTSTCTRLL, RSCANnGTSTCTRLH, RSCANnGTSTCTRHL, RSCANnGTSTCTRHH registers can be read/written in 8-bit units
Address(es):	RSCANnGTSTCTR: <RSCFDn_base> + H’046C RSCANnGTSTCTRL: <RSCFDn_base> + H’046C, RSCANnGTSTCTRH: <RSCFDn_base> + H’046E RSCANnGTSTCTRLL: <RSCFDn_base> + H’046C, RSCANnGTSTCTRLH: <RSCFDn_base> + H’046D, RSCANnGTSTCTRHL: <RSCFDn_base> + H’046E, RSCANnGTSTCTRHH: <RSCFDn_base> + H’046F
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	RTME	0	R/W	RAM Test Enable 0: RAM test is disabled. 1: RAM test is enabled.
1	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
0	ICBCTME	0	R/W	Communication Test between Channels Enable 0: Communication test between channels disabled 1: Communication test between channels enabled

RTME Bit

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode.

This bit is cleared to 0 in global reset mode.

1. Set the GMDC[1:0] bits in the RSCANnGCTR register to 10b (Global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0, 1) in the RSCANnGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode.

This bit is cleared to 0 in global reset mode.

28.3.14.3 RSCANnGLOCKK — Global Lock Key Register

Access Size:	RSCANnGLOCKK register can only be written in 32-bit units RSCANnGLOCKKL, RSCANnGLOCKKH registers can only be written in 16-bit units
Address(es):	RSCANnGLOCKK: <RSCFDn_base> + H’047C RSCANnGLOCKKL: <RSCFDn_base> + H’047C, RSCANnGLOCKKH: <RSCFDn_base> + H’047E
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]	W*[1]

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved When writing these bits, write the value after reset.
15 to 0	LOCK[15:0]	All 0	W*[1]	Lock Key These bits are key bits to release protection of test mode.

Note 1. Writing to these bits is effective only when the RS-CANFD module is in global test mode.

The RSCANnGLOCKK register releases protection of special test bits and is write only.

For the protection release data, see **Section 28.11.4.2, Procedure for Releasing the Protection**.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCANnGTSTCTR register.

After the protection has been released, writing to the I/O register area (<RSCFDn_base> + H’0000 to <RSCFDn_base> + H’04FF) of the CAN (except the RAM) enables the protection again.

Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.

28.3.14.4 RSCANnRPGACCr — RAM Test Page Access Register (r = 0 to 63)

Access Size: RSCANnRPGACCr register can be read/written in 32-bit units
RSCANnRPGACCrL, RSCANnRPGACCrH registers can be read/written in 16-bit units
RSCANnRPGACCrLL, RSCANnRPGACCrLH, RSCANnRPGACCrHL, RSCANnRPGACCrHH registers can be read/written in 8-bit units

Address(es): RSCANnRPGACCr: <RSCFDn_base> + H’1900 + (H’04 × r)
RSCANnRPGACCrL: <RSCFDn_base> + H’1900 + (H’04 × r),
RSCANnRPGACCrH: <RSCFDn_base> + H’1902 + (H’04 × r)
RSCANnRPGACCrLL: <RSCFDn_base> + H’1900 + (H’04 × r),
RSCANnRPGACCrLH: <RSCFDn_base> + H’1901 + (H’04 × r),
RSCANnRPGACCrHL: <RSCFDn_base> + H’1902 + (H’04 × r),
RSCANnRPGACCrHH: <RSCFDn_base> + H’1903 + (H’04 × r)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDTA[31:0]	All 0	R/W	RAM Data Test Access Data can be read and written in RSCAN RAM.

Modify the RSCANnRPGACCr register in global test mode with the RTME bit in the RSCANnGTSTCTR register set to 1 (RAM test is enabled).

The RSCANnRPGACCr register is readable and writable when the RTME bit is set to 1.

28.4 Registers (CANFD Mode)

This section describes all registers to be used when the RS-CANFD is used in CANFD mode.

28.4.1 List of Registers

The following tables list RS-CANFD registers to be used in CANFD mode.

For details about <RSCFDn_base>, see **Section 28.1.2, Register Base Address**.

- **Table 28.37, Registers Initialized in Global Reset Mode or Channel Reset Mode**
- **Table 28.38, Registers Initialized Only in Global Reset Mode**

Table 28.20 Registers (1/3)

Module	Register	Symbol	Address
Interface mode-related register			
RSCFDn	Global interface mode select register	RSCFDnCFDGRMCFG	<RSCFDn_base> + H'04FC
Channel-related registers			
RSCFDn	Channel m nominal bit rate configuration register	RSCFDnCFDCmNCFG	<RSCFDn_base> + H'0000 + (H'10 × m)
RSCFDn	Channel m control register	RSCFDnCFDCmCTR	<RSCFDn_base> + H'0004 + (H'10 × m)
RSCFDn	Channel m status register	RSCFDnCFDCmSTS	<RSCFDn_base> + H'0008 + (H'10 × m)
RSCFDn	Channel m error flag register	RSCFDnCFDCmERFL	<RSCFDn_base> + H'000C + (H'10 × m)
RSCFDn	Channel m data bit rate configuration register	RSCFDnCFDCmDCFG	<RSCFDn_base> + H'0500 + (H'20 × m)
RSCFDn	Channel m CANFD configuration register	RSCFDnCFDCmFDCFG	<RSCFDn_base> + H'0504 + (H'20 × m)
RSCFDn	Channel m CANFD control register	RSCFDnCFDCmFDCTR	<RSCFDn_base> + H'0508 + (H'20 × m)
RSCFDn	Channel m CANFD status register	RSCFDnCFDCmFDSTS	<RSCFDn_base> + H'050C + (H'20 × m)
RSCFDn	Channel m CANFD CRC register	RSCFDnCFDCmFDCRC	<RSCFDn_base> + H'0510 + (H'20 × m)
Global-related registers			
RSCFDn	Global configuration register	RSCFDnCFDGCFG	<RSCFDn_base> + H'0084
RSCFDn	Global control register	RSCFDnCFDGCTR	<RSCFDn_base> + H'0088
RSCFDn	Global status register	RSCFDnCFDGSTS	<RSCFDn_base> + H'008C
RSCFDn	Global error flag register	RSCFDnCFDGERFL	<RSCFDn_base> + H'0090
RSCFDn	Global timestamp counter register	RSCFDnCFDGTSC	<RSCFDn_base> + H'0094
RSCFDn	Global TX Interrupt Status Register 0	RSCFDnCFDGTINTSTS0	<RSCFDn_base> + H'0460
RSCFDn	Global FD configuration register	RSCFDnCFDGFDCFG	<RSCFDn_base> + H'0474
Receive rule-related registers			
RSCFDn	Receive Rule Entry Control Register	RSCFDnCFDGAFLECTR	<RSCFDn_base> + H'0098
RSCFDn	Receive Rule Configuration Register 0	RSCFDnCFDGAFLCFG0	<RSCFDn_base> + H'009C
RSCFDn	Receive Rule ID Register j	RSCFDnCFDGAFLIDj	<RSCFDn_base> + H'1000 + (H'10 × j)
RSCFDn	Receive Rule Mask Register j	RSCFDnCFDGAFLMj	<RSCFDn_base> + H'1004 + (H'10 × j)
RSCFDn	Receive Rule Pointer 0 Register j	RSCFDnCFDGAFLP0_j	<RSCFDn_base> + H'1008 + (H'10 × j)
RSCFDn	Receive Rule Pointer 1 Register j	RSCFDnCFDGAFLP1_j	<RSCFDn_base> + H'100C + (H'10 × j)

Table 28.20 Registers (2/3)

Module	Register	Symbol	Address
Receive buffer-related registers			
RSCFDn	Receive Buffer Number Register	RSCFDnCFDRMNB	<RSCFDn_base> + H'00A4
RSCFDn	Receive Buffer New Data Register y	RSCFDnCFDRMNDy	<RSCFDn_base> + H'00A8 + (H'04 × y)
RSCFDn	Receive Buffer ID Register q	RSCFDnCFDRMIDq	<RSCFDn_base> + H'2000 + (H'20 × q)
RSCFDn	Receive Buffer Pointer Register q	RSCFDnCFDRMPTRq	<RSCFDn_base> + H'2004 + (H'20 × q)
RSCFDn	Receive Buffer CANFD status register q	RSCFDnCFDRMFDSTSq	<RSCFDn_base> + H'2008 + (H'20 × q)
RSCFDn	Receive Buffer Data Field b Register q	RSCFDnCFDRMDFb_q	<RSCFDn_base> + H'200C + (H'04 × b) + (H'20 × q)
Receive FIFO buffer-related registers			
RSCFDn	Receive FIFO Buffer Configuration and Control Register x	RSCFDnCFDRFCCx	<RSCFDn_base> + H'00B8 + (H'04 × x)
RSCFDn	Receive FIFO Buffer Status Register x	RSCFDnCFDRFSTSx	<RSCFDn_base> + H'00D8+ (H'04 × x)
RSCFDn	Receive FIFO Buffer Pointer Control Register x	RSCFDnCFDRFPCTRx	<RSCFDn_base> + H'00F8 + (H'04 × x)
RSCFDn	Receive FIFO Buffer Access ID Register x	RSCFDnCFDRFIDx	<RSCFDn_base> + H'3000 + (H'80 × x)
RSCFDn	Receive FIFO Buffer Access Pointer Register x	RSCFDnCFDRFPTRx	<RSCFDn_base> + H'3004 + (H'80 × x)
RSCFDn	Receive FIFO CANFD status register x	RSCFDnCFDRFFDSTSx	<RSCFDn_base> + H'3008 + (H'80 × x)
RSCFDn	Receive FIFO Buffer Access Data Field d Register x	RSCFDnCFDRFDFd_x	<RSCFDn_base> + H'300C + (H'04 × d) + (H'80 × x)
Transmit/Receive FIFO buffer related registers			
RSCFDn	Transmit/receive FIFO Buffer Configuration and Control Register k	RSCFDnCFDCFCCk	<RSCFDn_base> + H'0118 + (H'04 × k)
RSCFDn	Transmit/receive FIFO Buffer Status Register k	RSCFDnCFDCFSTSk	<RSCFDn_base> + H'0178 + (H'04 × k)
RSCFDn	Transmit/receive FIFO Buffer Pointer Control Register k	RSCFDnCFDCFPCTRk	<RSCFDn_base> + H'01D8 + (H'04 × k)
RSCFDn	Transmit/receive FIFO Buffer Access ID Register k	RSCFDnCFDCFIDk	<RSCFDn_base> + H'3400 + (H'80 × k)
RSCFDn	Transmit/receive FIFO Buffer Access Pointer Register k	RSCFDnCFDCFPTRk	<RSCFDn_base> + H'3404 + (H'80 × k)
RSCFDn	Transmit/receive FIFO CANFD configuration/status register k	RSCFDnCFDCFFDCSTSk	<RSCFDn_base> + H'3408 + (H'80 × k)
RSCFDn	Transmit/receive FIFO Buffer Access Data Field d Register k	RSCFDnCFDCFDFd_k	<RSCFDn_base> + H'340 + (H'04 × d) + (H'80 × k)
FIFO status-related registers			
RSCFDn	FIFO Empty Status Register	RSCFDnCFDFESTS	<RSCFDn_base> + H'0238
RSCFDn	FIFO Full Status Register	RSCFDnCFDFFSTS	<RSCFDn_base> + H'023C
RSCFDn	FIFO Message Lost Status Register	RSCFDnCFDFMSTS	<RSCFDn_base> + H'0240
RSCFDn	Receive FIFO Buffer Interrupt Flag Status Register	RSCFDnCFDRFISTS	<RSCFDn_base> + H'0244
RSCFDn	Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register	RSCFDnCFDCFRISTS	<RSCFDn_base> + H'0248
RSCFDn	Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCFDnCFDCFTISTS	<RSCFDn_base> + H'024C
FIFO DMA-related registers			
RSCFDn	DMA enable register	RSCFDnCFDCDTCT	<RSCFDn_base> + H'0490
RSCFDn	DMA status register	RSCFDnCFDCDTSTS	<RSCFDn_base> + H'0494
Transmit buffer-related registers			
RSCFDn	Transmit Buffer Control Register p	RSCFDnCFDTMCp	<RSCFDn_base> + H'0250 + (H'01 × p)
RSCFDn	Transmit Buffer Status Register p	RSCFDnCFDTMSTSp	<RSCFDn_base> + H'02D0 + (H'01 × p)
RSCFDn	Transmit Buffer ID Register p	RSCFDnCFDTMIDp	<RSCFDn_base> + H'4000 + (H'20 × p)
RSCFDn	Transmit Buffer Pointer Register p	RSCFDnCFDTMPTRp	<RSCFDn_base> + H'4004 + (H'20 × p)
RSCFDn	Transmit Buffer CANFD configuration register p	RSCFDnCFDTMFDCTRp	<RSCFDn_base> + H'4008 + (H'20 × p)
RSCFDn	Transmit Buffer Data Field b Register p	RSCFDnCFDTMDFb_p	<RSCFDn_base> + H'400C + (H'04 × b) + (H'20 × p)
RSCFDn	Transmit Buffer Interrupt Enable Configuration Register y	RSCFDnCFDTMIECy	<RSCFDn_base> + H'0390 + (H'04 × y)

Table 28.20 Registers (3/3)

Module	Register	Symbol	Address
Transmit buffer status-related registers			
RSCFDn	Transmit Buffer Transmit Request Status Register y	RSCFDnCFDTMTRSTSy	<RSCFDn_base> + H'0350 + (H'04 × y)
RSCFDn	Transmit Buffer Transmit Abort Request Status Register y	RSCFDnCFDTMTARSTSy	<RSCFDn_base> + H'0360 + (H'04 × y)
RSCFDn	Transmit Buffer Transmit Complete Status Register y	RSCFDnCFDTMTCSTSy	<RSCFDn_base> + H'0370 + (H'04 × y)
RSCFDn	Transmit Buffer Transmit Abort Status Register y	RSCFDnCFDTMTASTSy	<RSCFDn_base> + H'0380 + (H'04 × y)
Transmit queue-related registers			
RSCFDn	Transmit Queue Configuration and Control Register m	RSCFDnCFDTXQCCm	<RSCFDn_base> + H'03A0 + (H'04 × m)
RSCFDn	Transmit Queue Status Register m	RSCFDnCFDTXQSTSm	<RSCFDn_base> + H'03C0 + (H'04 × m)
RSCFDn	Transmit Queue Pointer Control Register m	RSCFDnCFDTXQPCTRm	<RSCFDn_base> + H'03E0 + (H'04 × m)
Transmit history-related registers			
RSCFDn	Transmit History Configuration and Control Register m	RSCFDnCFDTHLCCm	<RSCFDn_base> + H'0400 + (H'04 × m)
RSCFDn	Transmit History Status Register m	RSCFDnCFDTHLSTSm	<RSCFDn_base> + H'0420 + (H'04 × m)
RSCFDn	Transmit History Pointer Control Register m	RSCFDnCFDTHLPCTRm	<RSCFDn_base> + H'0440 + (H'04 × m)
RSCFDn	Transmit History Access Register m	RSCFDnCFDTHLACCm	<RSCFDn_base> + H'6000 + (H'04 × m)
Test-related registers			
RSCFDn	Global Test Configuration Register	RSCFDnCFDGTSTCFG	<RSCFDn_base> + H'0468
RSCFDn	Global Test Control Register	RSCFDnCFDGTSTCTR	<RSCFDn_base> + H'046C
RSCFDn	Global Lock Key Register	RSCFDnCFDGLOCKK	<RSCFDn_base> + H'047C
RSCFDn	RAM Test Page Access Register r	RSCFDnCFDRPGACCr	<RSCFDn_base> + H'6400 + (H'04 × r)

Table 28.21 Transmit Buffer p Allocated to Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 28.22 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANm
Transmit buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 28.23 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000b	Transmit buffer 16 × m + 0
0001b	Transmit buffer 16 × m + 1
0010b	Transmit buffer 16 × m + 2
0011b	Transmit buffer 16 × m + 3
0100b	Transmit buffer 16 × m + 4
0101b	Transmit buffer 16 × m + 5
0110b	Transmit buffer 16 × m + 6
0111b	Transmit buffer 16 × m + 7
1000b	Transmit buffer 16 × m + 8
1001b	Transmit buffer 16 × m + 9
1010b	Transmit buffer 16 × m + 10
1011b	Transmit buffer 16 × m + 11
1100b	Transmit buffer 16 × m + 12
1101b	Transmit buffer 16 × m + 13
1110b	Transmit buffer 16 × m + 14
1111b	Transmit buffer 16 × m + 15

Table 28.24 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC[3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000b	Setting prohibited
0001b	Setting prohibited
0010b	Transmit buffer 16 × m + 15 to 16 × m + 13
0011b	Transmit buffer 16 × m + 15 to 16 × m + 12
0100b	Transmit buffer 16 × m + 15 to 16 × m + 11
0101b	Transmit buffer 16 × m + 15 to 16 × m + 10
0110b	Transmit buffer 16 × m + 15 to 16 × m + 9
0111b	Transmit buffer 16 × m + 15 to 16 × m + 8
1000b	Transmit buffer 16 × m + 15 to 16 × m + 7
1001b	Transmit buffer 16 × m + 15 to 16 × m + 6
1010b	Transmit buffer 16 × m + 15 to 16 × m + 5
1011b	Transmit buffer 16 × m + 15 to 16 × m + 4
1100b	Transmit buffer 16 × m + 15 to 16 × m + 3
1101b	Transmit buffer 16 × m + 15 to 16 × m + 2
1110b	Transmit buffer 16 × m + 15 to 16 × m + 1
1111b	Transmit buffer 16 × m + 15 to 16 × m + 0

28.4.2 Details of Interface Mode-related Registers

28.4.2.1 RSCFDnCFDGRMCFG — Global Interface Mode Select Register

Access Size:	RSCFDnCFDGRMCFG register can be read/written in 32-bit units RSCFDnCFDGRMCFGL, RSCFDnCFDGRMCFGH registers can be read/written in 16-bit units RSCFDnCFDGRMCFGLL, RSCFDnCFDGRMCFGLH, RSCFDnCFDGRMCFGHL, RSCFDnCFDGRMCFGHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGRMCFG: <RSCFDn_base> + H’04FC RSCFDnCFDGRMCFGL: <RSCFDn_base> + H’04FC, RSCFDnCFDGRMCFGH: <RSCFDn_base> + H’04FE RSCFDnCFDGRMCFGLL: <RSCFDn_base> + H’04FC, RSCFDnCFDGRMCFGLH: <RSCFDn_base> + H’04FD, RSCFDnCFDGRMCFGHL: <RSCFDn_base> + H’04FE, RSCFDnCFDGRMCFGHH: <RSCFDn_base> + H’04FF
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RCMC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
0	RCMC	0	R/W	Interface Mode Select 0: Classical CAN mode 1: CANFD mode

Note: The RSCANnGRMCFG register and the RSCFDnCFDGRMCFG register are the same register. Therefore, set either one of the registers.

Modify the RSCFDnCFDGRMCFG register only in global reset mode. Before setting other RS-CANFD registers, set this register.

RCMC Bit

Setting this bit to 0 places the RS-CANFD module in classical CAN mode. Setting this bit to 1 places the RS-CANFD module in CANFD mode. To switch classical CAN mode to CANFD mode, set the value after reset in all registers and bits allocated to the register map of classical CAN mode and then modify the RSCFDnCFDGRMCFG register.

28.4.3 Details of Channel-related Registers

28.4.3.1 RSCFDnCFDCmNCFG — Channel Nominal Bit Rate Configuration Register (m = 0, 1)

Access Size: RSCFDnCFDCmNCFG register can be read/written in 32-bit units
RSCFDnCFDCmNCFGL, RSCFDnCFDCmNCFGH registers can be read/written in 16-bit units
RSCFDnCFDCmNCFGLL, RSCFDnCFDCmNCFGLH, RSCFDnCFDCmNCFGHL, RSCFDnCFDCmNCFGHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDCmNCFG: <RSCFDn_base> + H’0000 + (H’10 × m)
RSCFDnCFDCmNCFGL: <RSCFDn_base> + H’0000 + (H’10 × m),
RSCFDnCFDCmNCFGH: <RSCFDn_base> + H’0002 + (H’10 × m)
RSCFDnCFDCmNCFGLL: <RSCFDn_base> + H’0000 + (H’10 × m),
RSCFDnCFDCmNCFGLH: <RSCFDn_base> + H’0001 + (H’10 × m),
RSCFDnCFDCmNCFGHL: <RSCFDn_base> + H’0002 + (H’10× m),
RSCFDnCFDCmNCFGHH: <RSCFDn_base> + H’0003 + (H’10 × m)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	NTSEG2[4:0]					—	NTSEG1[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NSJW[4:0]					—	NBRP[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
28 to 24	NTSEG2[4:0]	All 0	R/W	Nominal Bit Rate Time Segment 2 Control b28 b27 b26 b25 b24 0 0 0 0 0: Setting prohibited 0 0 0 0 1: 2 Tq : : 1 1 1 1 0: 31 Tq 1 1 1 1 1: 32 Tq
23	—	0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Bit	Bit Name	Initial Value	R/W	Description
22 to 16	NTSEG1[6:0]	All 0	R/W	Nominal Bit Rate Time Segment 1 Control b22 b21 b20 b19 b18 b17 b16 0 0 0 0 0 0 0: Setting prohibited 0 0 0 0 0 0 1: Setting prohibited 0 0 0 0 0 1 0: Setting prohibited 0 0 0 0 0 1 1: 4 Tq : : 1 1 1 1 1 1 0: 127 Tq 1 1 1 1 1 1 1: 128 Tq
15 to 11	NSJW[4:0]	All 0	R/W	Nominal Bit Rate Resynchronization Jump Width Control b15 b14 b13 b12 b11 0 0 0 0 0: 1 Tq 0 0 0 0 1: 2 Tq 0 0 0 1 0: 3 Tq : : 1 1 1 1 0: 31 Tq 1 1 1 1 1: 32 Tq
10	—	0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
9 to 0	NBRP[9:0]	All 0	R/W	Nominal Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 1023), the nominal bit rate prescaler divides fCAN by (P + 1).

Modify the RSCFDnCFDCmNCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode, and then transition to channel communication mode or channel halt mode. For the description and settings of bit timing parameters, see **Section 28.11.1, Initial Settings**.

NTSEG2[4:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) of nominal bit rate. Allowed values are 2 Tq to 32 Tq, inclusive.

Set a value smaller than the value of the NTSEG1[6:0] bits.

NTSEG1[6:0] Bits

These bits specify the total length of propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) of nominal bit rate as a Tq value.

A value of 4 to 128 Tq is settable.

NSJW[4:0] Bits

These bits specify the resynchronization jump width of nominal bit rate as a Tq value. A value of 1 to 32 Tq is settable. Specify a value equal to or smaller than the NTSEG2[4:0] value.

NBRP[9:0] Bits

The clock obtained by dividing the CAN clock (fCAN) by the nominal bit rate prescaler ((NBRP[9:0]) + 1) becomes CANmTq(N) clock (fCANTQ(N)m). One clock of the CANmTq(N) clock becomes one Time Quantum (Tq).

28.4.3.2 RSCFDnCFDCmCTR — Channel Control Register (m = 0, 1)

Access Size: RSCFDnCFDCmCTR register can be read/written in 32-bit units
RSCFDnCFDCmCTRL, RSCFDnCFDCmCTRH registers can be read/written in 16-bit units
RSCFDnCFDCmCTRLL, RSCFDnCFDCmCTRLH, RSCFDnCFDCmCTRHL, RSCFDnCFDCmCTRHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDCmCTR: <RSCFDn_base> + H’0004 + (H’10 × m)
RSCFDnCFDCmCTRL: <RSCFDn_base> + H’0004 + (H’10 × m),
RSCFDnCFDCmCTRH: <RSCFDn_base> + H’0006 + (H’10 × m)
RSCFDnCFDCmCTRLL: <RSCFDn_base> + H’0004 + (H’10 × m),
RSCFDnCFDCmCTRLH: <RSCFDn_base> + H’0005 + (H’10 × m),
RSCFDnCFDCmCTRHL: <RSCFDn_base> + H’0006 + (H’10 × m),
RSCFDnCFDCmCTRHH: <RSCFDn_base> + H’0007 + (H’10 × m)

Initial Value: H’0000 0005

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ROM	CRCT	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	TDCVFIE	SOCOIE	EOCOIE	TAIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	ROM	0	R/W	Restricted Operation Mode Enable 0: Restricted operation mode is disabled. 1: Restricted operation mode is enabled.
30	CRCT	0	R/W	CRC Error Test Enable 0: The first bit of the reception ID field is not inverted. 1: The first bit of the reception ID field is inverted.
29 to 27	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
26, 25	CTMS[1:0]	All 0	R/W	Communication Test Mode Select b26 b25 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	0	R/W	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	0	R/W	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits 14 to 8 in the RSCFDnCFDCmERFL register are all cleared. 1: Error flags for all error information are displayed.

Bit	Bit Name	Initial Value	R/W	Description
22, 21	BOM[1:0]	All 0	R/W	Bus Off Recovery Mode Select b22 b21 0 0: ISO11898-1 compliant 0 1: Entry to channel halt mode automatically at bus-off entry 1 0: Entry to channel halt mode automatically at bus-off end 1 1: Entry to channel halt mode (in bus-off state) by program request
20	—	0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
19	TDCVFIE	0	R/W	Transmitter Delay Compensation Violation Interrupt Enable 0: A transmitter delay compensation violation interrupt is disabled. 1: A transmitter delay compensation violation interrupt is enabled.
18	SOCOIE	0	R/W	Successful Occurrence Counter Overflow Interrupt Enable 0: A successful occurrence counter overflow interrupt is disabled. 1: A successful occurrence counter overflow interrupt is enabled.
17	EOCOIE	0	R/W	Error Occurrence Counter Overflow Interrupt Enable 0: An error occurrence counter overflow interrupt is disabled. 1: An error occurrence counter overflow interrupt is enabled.
16	TAIE	0	R/W	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	0	R/W	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	0	R/W	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	0	R/W	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.
12	BORIE	0	R/W	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	0	R/W	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	0	R/W	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	0	R/W	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	0	R/W	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
3	RTBO	0	R/W	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.

Bit	Bit Name	Initial Value	R/W	Description
2	CSLPR	1	R/W	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	01b	R/W	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

ROM Bit

When the ROM bit and the CTME bit in the RSCFDnCFDCmCTR register are set to 1, restricted operation mode is enabled. Use the restricted operation mode only when the CTMS[1:0] value in the RSCFDnCFDCmCTR register is 00b (standard test mode). Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CRCT Bit

This bit is used to test the CRC generation circuit in the RS-CANFD module. Setting this bit to 1 inverts the first bit of the ID field when a message is received. With this inversion of bit, the CRC calculation result does not match the normal CRC value of the received frame, which can detect a CRC error (the CERR bit in the RSCFDnCFDCmERFL register is 1). When using this function, note the following.

- This function is available while the CTME bit in the RSCFDnCFDCmCTR register is 1 (communication test mode enabled).
- This function cannot communicate with other CAN nodes. Use this function for inter-channel communication test (the CmICBCE bit in the RSCFDnCFDGTSTCFG register is 1).
- Bit inversion in the ID field may cause bit stuffing rule violation. In that case, no CRC error is detected but a stuff error is detected.

Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CTMS[1:0] Bits

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control the display mode of bits 14 to 8 in the RSCFDnCFDCmERFL register.

When this bit is clear to 0, if any error is detected while the flags of bits 14-8 in the RSCFDnCFDCmERFL register are all 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.

When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order. Modify this bit only in channel reset mode or channel halt mode.

BOM[1:0] Bits

These bits are used to select the bus off recovery mode of the RS-CANFD module.

When the BOM[1:0] bits are set to 00b, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10b (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 01b, the CHMDC[1:0] bits in the RSCFDnCFDCmCTR register (m = 0, 1) are set to 10b and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register are cleared to H’00.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 10b, the CHMDC[1:0] bits are set to 10b and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H’00.

When the BOM[1:0] bits are set to 11b and the CHMDC[1:0] bits are set to 10b while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H’00. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10b, a bus off recovery interrupt request is generated.

If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01b or at bus off end when the BOM[1:0] bits are 10b), the program’s writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.

TDCVFIE Bit

When the TDCVF flag in the RSCFDnCFDCmFDSTS register is set to 1 after the TDCVFIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

SOCOIE Bit

When the SOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 after the SOCOIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

EOCOIE Bit

When the EOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 after the EOCOIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

TAIE Bit

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.

ALIE Bit

When the ALF flag in the RSCFDnCFDCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BLIE Bit

When the BLF flag in the RSCFDnCFDCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCFDnCFDCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BORIE Bit

When the BORF flag in the RSCFDnCFDCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EPIE Bit

When the EPF flag in the RSCFDnCFDCmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EWIE Bit

When the EWF flag in the RSCFDnCFDCmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BEIE Bit

When the BEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

RTBO Bit

Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register to H’00 and also clears the BOSTS flag in the RSCFDnCFDCmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCFDnCFDCmCTR register are 00b (ISO11898-1 compliant).

A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RSCAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel into channel stop mode.

Clearing this bit to 0 makes the channel exit channel stop mode.

This bit should not be modified in channel communication mode or channel halt mode.

CHMDC[1:0] Bits

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see **Figure 28.5**. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode.

Do not set the CHMDC[1:0] bits to 11b. When the CAN module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become 10b.

28.4.3.3 RSCFDnCFDCmSTS — Channel Status Register (m = 0, 1)

Access Size:	RSCFDnCFDCmSTS register can be read/written in 32-bit units RSCFDnCFDCmSTSL, RSCFDnCFDCmSTSH registers can be read/written in 16-bit units RSCFDnCFDCmSTSLL, RSCFDnCFDCmSTSLH, RSCFDnCFDCmSTSHL, RSCFDnCFDCmSTSHH registers can be read/written in 8- bit units
Address(es):	RSCFDnCFDCmSTS: <RSCFDn_base> + H'0008 + (H'10 × m) RSCFDnCFDCmSTSL: <RSCFDn_base> + H'0008 + (H'10 × m), RSCFDnCFDCmSTSH: <RSCFDn_base> + H'000A + (H'10 × m) RSCFDnCFDCmSTSLL: <RSCFDn_base> + H'0008 + (H'10 × m), RSCFDnCFDCmSTSLH: <RSCFDn_base> + H'0009 + (H'10 × m), RSCFDnCFDCmSTSHL: <RSCFDn_base> + H'000A + (H'10 × m), RSCFDnCFDCmSTSHH: <RSCFDn_base> + H'000B + (H'10 × m)
Initial Value:	H'0000 0005

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ESIF	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R/W*1	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TEC[7:0]	All 0	R	The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	All 0	R	The receive error counter (REC) can be read.
15 to 9	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
8	ESIF	0	R/W*1	Error State Indication Flag 0: No CANFD message whose ESI bit is recessive has been received. 1: At least one CANFD message whose ESI bit is recessive has been received.
7	COMSTS	0	R	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	0	R	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	0	R	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	0	R	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	0	R	Error Passive Status Flag 0: Not in error passive state 1: In error passive state

Bit	Bit Name	Initial Value	R/W	Description
2	CSLPSTS	1	R	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	0	R	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode
0	CRSTSTS	1	R	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

ESIF Flag

When the recessive ESI bit is detected in a successfully received message, this flag is set to 1. In loopback mode or mirror mode, the own transmission message is regarded as a received message. To clear this flag to 0, write 0 to this bit by the program. This bit cannot be set to 1 by the program. If the flag setting (to 1) timing matches the writing 0 (by the program) timing, this flag is set to 1.

This flag is 0 in channel reset mode.

COMSTS Flag

This bit indicates that communication is ready.

This flag becomes 1 when the CAN module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the bus off state (TEC[7:0] > 255) is entered. It is cleared to 0 when the CAN module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the RS-CANFD module has entered the error passive state ((128 ≤ TEC[7:0] ≤ 255) or (128 ≤ REC[7:0])), It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel stop mode, and is cleared to 0 when the CAN module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel halt mode, and is cleared to 0 when the CAN module has returned from channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel reset mode, and is cleared to 0 when the CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the CAN module transitions from channel reset mode to channel stop mode.

28.4.3.4 RSCFDnCFDCmERFL — Channel Error Flag Register (m = 0, 1)

Access Size:	RSCFDnCFDCmERFL register can be read/written in 32-bit units RSCFDnCFDCmERFLL, RSCFDnCFDCmERFLH registers can be read/written in 16-bit units RSCFDnCFDCmERFLLL, RSCFDnCFDCmERFLLH, RSCFDnCFDCmERFLHL, RSCFDnCFDCmERFLHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDCmERFL: <RSCFDn_base> + H'000C + (H'10 × m) RSCFDnCFDCmERFLL: <RSCFDn_base> + H'000C + (H'10 × m), RSCFDnCFDCmERFLH: <RSCFDn_base> + H'000E + (H'10 × m) RSCFDnCFDCmERFLLL: <RSCFDn_base> + H'000C + (H'10 × m), RSCFDnCFDCmERFLLH: <RSCFDn_base> + H'000D + (H'10 × m), RSCFDnCFDCmERFLHL: <RSCFDn_base> + H'000E + (H'10 × m), RSCFDnCFDCmERFLHH: <RSCFDn_base> + H'000F + (H'10 × m)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
30 to 16	CRCREG [14:0]	All 0	R	CRC Calculation Data (CRC length:15 bits) A CRC value calculated based on the transmit message or receive message is indicated.
15	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
14	ADERR	0	R/W*1	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	0	R/W*1	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	0	R/W*1	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	0	R/W*1	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.
10	AERR	0	R/W*1	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.

Bit	Bit Name	Initial Value	R/W	Description
9	FERR	0	R/W*1	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	0	R/W*1	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	0	R/W*1	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.
6	BLF	0	R/W*1	Bus Lock Flag 0: No channel bus is detected. 1: Channel bus is detected.
5	OVLF	0	R/W*1	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	0	R/W*1	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	0	R/W*1	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	0	R/W*1	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	0	R/W*1	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	0	R/W*1	Bus Error Flag 0: No channel bus error is detected. 1: Channel bus error is detected.

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.

If the ERRD bit in the RSCFDnCFDCmCTR register is set to 0 (i.e., only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCFDnCFDCmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the when time the error occurred.

CRCREG[14:0] Flag

When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode is enabled), if transmit or receive message is a classical CAN frame (CRC length = 15 bits), this flag is updated and the CRC value calculated based on the message can be read. When a CANFD frame is sent or received, the value of CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register is updated. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration-lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.

- A recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- The CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0.

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the CAN module returns from the bus off state. However, this flag is not set to 1 if the CAN module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to 01b (channel reset mode).
- The RTBO bit in the RSCFDnCFDCmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCFDnCFDCmCTR register are set to 01b (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to 10b (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11b (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCFDnCFDCmCTR register (m = 0, 1) set to 01b (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127).

This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCFDnCFDCmERFL register is set to 1.

NOTE

To clear the flag of this register to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

28.4.3.5 RSCFDnCFDCmDCFG — Channel Data Bit Rate Configuration Register (m = 0, 1)

Access Size: RSCFDnCFDCmDCFG register can be read/written in 32-bit units

RSCFDnCFDCmDCFGL, RSCFDnCFDCmDCFGH registers can be read/written in 16-bit units

RSCFDnCFDCmDCFGLL, RSCFDnCFDCmDCFGLH, RSCFDnCFDCmDCFGHL, RSCFDnCFDCmDCFGHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDCmDCFG: <RSCFDn_base> + H’0500 + (H’20 × m)

RSCFDnCFDCmDCFGL: <RSCFDn_base> + H’0500 + (H’20 × m),

RSCFDnCFDCmDCFGH: <RSCFDn_base> + H’0502 + (H’20 × m)

RSCFDnCFDCmDCFGLL: <RSCFDn_base> + H’0500 + (H’20 × m),

RSCFDnCFDCmDCFGLH: <RSCFDn_base> + H’0501 + (H’20 × m),

RSCFDnCFDCmDCFGHL: <RSCFDn_base> + H’0502 + (H’20 × m),

RSCFDnCFDCmDCFGHH: <RSCFDn_base> + H’0503 + (H’20 × m)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	DSJW[2:0]			—	DTSEG2[2:0]			DTSEG1[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DBRP[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
26 to 24	DSJW[2:0]	All 0	R/W	Data Bit Rate Resynchronization Jump Width Control b26 b25 b24 0 0 0: 1 Tq 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq
23	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
22 to 20	DTSEG2[2:0]	All 0	R/W	Data Bit Rate Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq

Bit	Bit Name	Initial Value	R/W	Description
19 to 16	DTSEG1[3:0]	All 0	R/W	Data Bit Rate Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: 2 Tq 0 0 1 0: 3 Tq 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 8	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
7 to 0	DBRP[7:0]	All 0	R/W	Data Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 255), the data bit rate prescaler divides fCAN by (P + 1).

Modify the RSCFDnCFDCmDCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode, and then transition to channel communication mode or channel halt mode. When only classical CAN frames are used in CANFD mode, set the RSCFDnCFDCmDCFG register to the value equal to the set RSCFDnCFDCmNCFG register value. For the description and settings of bit timing parameters, see **Section 28.11.1, Initial Settings**.

DSJW[2:0] Bits

These bits specify the resynchronization jump width of data bit rate as a Tq value. A value of 1 to 8 Tq is settable. Specify a value equal to or smaller than the DTSEG2[2:0] bits value.

DTSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) of nominal bit rate. Allowed values are 2 Tq to 8 Tq, inclusive.

Set a value smaller than the value of the DTSEG1[3:0] bits.

DTSEG1[3:0] Bits

These bits specify the total length of propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) of data bit rate as a Tq value.

A value of 2 to 16 Tq is settable.

DBRP[7:0] Bits

The clock obtained by dividing the CAN clock (fCAN) by the data bit rate prescaler ((DBRP[7:0]) + 1) becomes CANmTq(D) clock (fCANTQ(D)m). One clock of the CANmTq(D) clock becomes one Time Quantum (Tq).

Make sure to set the same value in the NBRP[9:0] bits and the DBRP[7:0] bits. To set the nominal bit rate and data bit rate of different values, change the NTSEG 1 bit and NTSEG 2 bit in the RSCFDnCFDCmNCFG register and the DTSEG 1 bit and DTSEG 2 bit in the RSCFDnCFDCmDCFG register to the desired bit rate values respectively. When the TDCE bit of the RSCFDnCFDCmFDCFG register is 1 (transmission delay correction enabled), set the same value of 1 or less to the NBRP [9:0] and DBRP [7:0] bits.

28.4.3.6 RSCFDnCFDCmFDCFG — Channel CANFD Configuration Register (m = 0, 1)

Access Size: RSCFDnCFDCmFDCFG register can be read/written in 32-bit units

RSCFDnCFDCmFDCFGL, RSCFDnCFDCmFDCFGH registers can be read/written in 16-bit units

RSCFDnCFDCmFDCFGLL, RSCFDnCFDCmFDCFGLH, RSCFDnCFDCmFDCFGHL, RSCFDnCFDCmFDCFGHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDCmFDCFG: <RSCFDn_base> + H'0504 + (H'20 × m)

RSCFDnCFDCmFDCFGL: <RSCFDn_base> + H'0504 + (H'20 × m),

RSCFDnCFDCmFDCFGH: <RSCFDn_base> + H'0506 + (H'20 × m)

RSCFDnCFDCmFDCFGLL: <RSCFDn_base> + H'0504 + (H'20 × m),

RSCFDnCFDCmFDCFGLH: <RSCFDn_base> + H'0505 + (H'20 × m),

RSCFDnCFDCmFDCFGHL: <RSCFDn_base> + H'0506 + (H'20 × m),

RSCFDnCFDCmFDCFGHH: <RSCFDn_base> + H'0507 + (H'20 × m)

Initial Value: H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	REFE	FDOE	TMME	GWBRS	GWFDF	GWEN	—	TDCO[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ESIC	TDCE	TDCOC	—	—	—	—	—	EOCCFG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
29	REFE	0	R/W	Reception data edge filter enable bit 0: Reception data edge filter disabled 1: Reception data edge filter enabled
28	FDOE	0	R/W	FD-only mode enable bit 0: FD-only mode disabled 1: FD-only mode enabled
27	TMME	0	R/W	Transmit Buffer Merge Mode Enable 0: Transmit buffer merge mode is disabled. 1: Transmit buffer merge mode is enabled.
26	GWBRS	0	R/W	Gateway BRS 0: A frame is transmitted with the BRS bit in the received frame set to 0. 1: A frame is transmitted with the BRS bit in the received frame set to 1.
25	GWFDF	0	R/W	Gateway FDF 0: A frame is transmitted regarding the received frame as a classical CAN frame. 1: A frame is transmitted regarding the received frame as a CANFD frame.
24	GWEN	0	R/W	CAN-CANFD Gateway Enable 0: The CAN-CANFD gateway is disabled. 1: The CAN-CANFD gateway is enabled.
23	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.

Bit	Bit Name	Initial Value	R/W	Description
22 to 16	TDCO[6:0]	All 0	R/W	Transmitter Delay Compensation Offset These bits are set to the transmitter delay compensation offset value.
15 to 11	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
10	ESIC	0	R/W	Error State Display Mode Select 0: Always displays the node error state. 1: When the node is not in the error passive state: Displays the message buffer error state. When the node is in the error passive state: Displays the node error state.
9	TDCE	0	R/W	Transmitter Delay Compensation Enable 0: Transmitter delay compensation is disabled. 1: Transmitter delay compensation is enabled.
8	TDCOC	0	R/W	Transmitter Delay Compensation Measurement Select 0: Measurement and offset 1: Only offset
7 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2 to 0	EOCCFG[2:0]	All 0	R/W	Error Occurrence Counting Method Select b2 b1 b0 0 0 0: All transmit messages and receive messages 0 0 1: All transmit messages 0 1 0: All receive messages 0 1 1: Setting prohibited 1 0 0: Only data phase of transmitted or received CANFD message 1 0 1: Only data phase of transmitted CANFD message 1 1 0: Only data phase of received CANFD message 1 1 1: Setting prohibited

REFE bit

Setting this bit to 1 enables reception data edge filtering when the idle condition is detected, and a dominant level with less than 2 time quanta is ignored. A dominant level with more than or equal to 2 time quanta is detected as an edge. Modify this bit only in channel reset mode.

FDOE bit

Setting this bit to 1 enables FD-only mode. When data is transmitted, a CANFD frame will be sent regardless of the settings to the CFFDF bit in the RSCFDnCFDCFFDCSTSk register or the TMFDF bit in the RSCFDnCFDTMFDCTRp register. When a Classical CAN frame is received, a form error is detected. Modify this bit only in channel reset mode.

TMME Bit

Setting this bit to 1 enables transmit buffer merge mode. Modify this bit only in channel reset mode or channel halt mode.

GWBRS Bit

When the GWEN bit is 1, the BRS bit in a CANFD frame to be transmitted by the gateway function is set. Write 0 to this bit to clear the GWBRS bit to 0. Modify this bit only in channel reset mode.

GWFDF Bit

When the GWEN bit is 1, the FDF bit in a CANFD frame to be transmitted by the gateway function is set. Modify this bit only in channel reset mode.

GWEN Bit

This bit is used to control the operation of the transmit/receive FIFO buffer with the CFM[1:0] bits in the RSCFDnCFDCFCCk register set to 10b (gateway mode).

Setting this bit to 1 enables the CAN-CANFD gateway, enabling transmission in a format different from that of frames received by the gateway function. Received frames are replaced in accordance with the settings of the GWFDF bit and the GWBRS bit. When the DLC value in the received classical CAN frame is 1001b or more and the GWFDF bit is 1 (CANFD frame), the DLC value is replaced with 1000b.

While this bit is 1, do not perform routing the following frames by using the gateway function.

- CANFD frames with a payload length of more than 8 bytes
- Remote frames

While this bit is 1, the following frame should be transmitted in the channel by setting of GWFDF.

- When GWFDF bit is set to 0, only classical CAN frames should be transmitted.
- When GWFDF bit is set to 1, only CANFD frames should be transmitted.

Table 28.25 shows the settings and formats of transmit frame and receive frame while the CAN-CANFD gateway is enabled.

Table 28.25 Operation when the CAN-CANFD Gateway is Enabled

Receive Frame				Transmit Frame		
Format	BRS Bit	Received DLC Value	GWFDF Bit	Format	BRS Bit	DLC Value to be Transmitted
Classical CAN	None	DLC ≤ 1000b	0	Classical CAN	None	Not replaced
		DLC > 1000b				
CANFD	Arbitrary	DLC ≤ 1000b				
Classical CAN	None	DLC ≤ 1000b	1	CANFD	According to GWBRS bit setting	Not replaced
		DLC > 1000b				Replaced with 1000b
CANFD	Arbitrary	DLC ≤ 1000b				Not replaced

TDCO[6:0] Bits

These bits are set to the SSP offset value. How to use this value depends on the TDCOC bit in the RSCFDnCFDCmFDCFG register.

When the TDCOC bit is 0, the transmitter delay compensation result equals to the total value of the measured delay value and the TDCO[6:0] value (rounded to the nearest integer Tq).

When the TDCOC bit is 1, the transmitter delay compensation result equals to the TDCO[6:0] value. The SSP offset value = (set value of TDCO[6:0] bits + 1).

Modify these bits only in channel reset mode or channel halt mode.

ESIC Bit

When the ESIC bit is set to 1, if the channel is in the error active state, the ESI bit value (CFESI bit in the RSCFDnCFDCFFDCSTSk register or TMESI bit in the RSCFDnCFDTMFDCTRp register) set in the transmit/receive FIFO buffer or transmit buffer is transmitted as an ESI bit value of the transmit message. When the channel is in the error passive state or the ESIC bit is 0, the channel status is transmitted as an ESI bit value. Modify this bit only in channel reset mode or channel halt mode.

Table 28.26 ESI Value to Be Transmitted

ESIC Bit	Channel Status	ESI Value to be Transmitted
0	Error active	0 (error active node)
	Error passive	1 (error passive node)
1	Error active	ESI value set in the transmit/receive FIFO buffer or transmit buffer CFESI bit in the RSCFDnCFDCFFDCSTSk register or TMESI bit in the RSCFDnCFDTMFDCTRp register)
	Error passive	1 (error passive node)

TDCE Bit

Setting this bit to 1 enables transmitter delay compensation. Modify this bit only in channel reset mode or channel halt mode.

TDCOC Bit

When this bit is 0, the SSP position is defined by the total of the measured delay value and the SSP offset value (fixed value).

When this bit is 1, the SSP position is defined only by the SSP offset value. Modify this bit only in channel reset mode or channel halt mode.

EOCCFG[2:0] Bits

These bits are used to select a frame format and a transmission/reception direction when the error occurrence counter counts CAN bus errors.

Modify these bits only in channel reset mode or channel halt mode.

28.4.3.7 RSCFDnCFDCmFDCTR — Channel CANFD Control Register (m = 0, 1)

Access Size:	RSCFDnCFDCmFDCTR register can be read/written in 32-bit units RSCFDnCFDCmFDCTRL, RSCFDnCFDCmFDCTRH registers can be read/written in 16-bit units RSCFDnCFDCmFDCTRLL, RSCFDnCFDCmFDCTRLH, RSCFDnCFDCmFDCTRHL, RSCFDnCFDCmFDCTRHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDCmFDCTR: <RSCFDn_base> + H’0508 + (H’20 × m) RSCFDnCFDCmFDCTRL: <RSCFDn_base> + H’0508 + (H’20 × m), RSCFDnCFDCmFDCTRH: <RSCFDn_base> + H’050A + (H’20 × m) RSCFDnCFDCmFDCTRLL: <RSCFDn_base> + H’0508 + (H’20 × m), RSCFDnCFDCmFDCTRLH: <RSCFDn_base> + H’0509 + (H’20 × m), RSCFDnCFDCmFDCTRHL: <RSCFDn_base> + H’050A + (H’20 × m), RSCFDnCFDCmFDCTRHH: <RSCFDn_base> + H’050B + (H’20 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOCCLR	EOCCLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
1	SOCCLR	0	R/W	Successful Occurrence Counter Clear Setting the SOCCLR bit to 1 clears the successful occurrence counter. This bit is always read as 0.
0	EOCCLR	0	R/W	Error Occurrence Counter Clear Setting the EOCCLR bit to 1 clears the error occurrence counter. This bit is always read as 0.

SOCCLR Bit

Setting this bit to 1 clears the successful occurrence counter (SOC[7:0] bits in the RSCFDnCFDCmFDSTS register). This bit is automatically cleared to 0.

EOCCLR Bit

Setting this bit to 1 clears the error occurrence counter (EOC[7:0] bits in the RSCFDnCFDCmFDSTS register). This bit is automatically cleared to 0.

28.4.3.8 RSCFDnCFDCmFDSTS — Channel CANFD Status Register (m = 0, 1)

Access Size:	RSCFDnCFDCmFDSTS register can be read/written in 32-bit units RSCFDnCFDCmFDSTSL, RSCFDnCFDCmFDSTSH registers can be read/written in 16-bit units RSCFDnCFDCmFDSTSLL, RSCFDnCFDCmFDSTSLH, RSCFDnCFDCmFDSTSHL, RSCFDnCFDCmFDSTSHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDCmFDSTS: <RSCFDn_base> + H’050C + (H’20 × m) RSCFDnCFDCmFDSTSL: <RSCFDn_base> + H’050C + (H’20 × m), RSCFDnCFDCmFDSTSH: <RSCFDn_base> + H’050E + (H’20 × m) RSCFDnCFDCmFDSTSLL: <RSCFDn_base> + H’050C + (H’20 × m), RSCFDnCFDCmFDSTSLH: <RSCFDn_base> + H’050D + (H’20 × m), RSCFDnCFDCmFDSTSHL: <RSCFDn_base> + H’050E + (H’20 × m), RSCFDnCFDCmFDSTSHH: <RSCFDn_base> + H’050F + (H’20 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SOC[7:0]								EOC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SOCO	EOCO	TDCVF	TDCR[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SOC[7:0]	All 0	R	Successful Occurrence Counter The successful occurrence counter value can be read.
23 to 16	EOC[7:0]	All 0	R	Error Occurrence Counter The error occurrence counter value can be read.
15 to 10	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
9	SOCO	0	R/W*1	Successful Occurrence Counter Overflow Flag 0: The successful occurrence counter does not overflow. 1: The successful occurrence counter has overflowed.
8	EOCO	0	R/W*1	Error Occurrence Counter Overflow Flag 0: The error occurrence counter does not overflow. 1: The error occurrence counter has overflowed.
7	TDCVF	0	R/W*1	Transmitter Delay Compensation Violation Flag 0: No transmitter delay compensation violation is present. 1: A transmitter delay compensation violation is present.
6 to 0	TDCR[6:0]	All 0	R	Transmitter Delay Compensation Result Status The transmitter delay compensation result can be read.

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

SOC[7:0] Bits

These bits show the successful occurrence counter value. The successful occurrence counter is incremented upon completion of message reception or transmission without an error. This counter stops counting when it reaches H’FF. In loopback mode, this counter is incremented twice.

These bits are cleared to 0 by writing 1 to the SOCCLR bit in the RSCFDnCFDCmCTR register. These bits are 0 in channel reset mode.

EOC[7:0] Bits

These bits show the error occurrence counter value. The error occurrence counter is incremented each time an error occurs according to the condition specified by the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register. This counter stops counting when it reaches H’FF.

These bits are cleared to 0 by writing 1 to the EOCCLR bit in the RSCFDnCFDCmCTR register. These bits are 0 in channel reset mode.

SOCO Flag

This bit indicates that successful occurrence counter overflow has occurred.

This flag is set to 1 when message reception or transmission is completed while the SOC[7:0] value has reached H’FF. This flag is 0 in channel reset mode.

EOCO Flag

This bit indicates that error occurrence counter overflow has occurred.

This flag is set to 1 when a CAN bus error is detected under the condition specified by the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register when the EOC[7:0] value has reached H’FF. This flag is 0 in channel reset mode.

TDCVF Flag

This bit indicates violation of transmitter delay compensation.

The transmit data is compared with the reception CAN bus level delayed due to the transceiver’s loop delay. This delay changes due to physical factors such as temperature. Because the TDCR[6:0] flags are updated for each message, temporary maximum delay cannot be confirmed.

This bit is set to 1 when the transmitter delay compensation exceeds the maximum compensation 3 CANm bit times - 2 Tq (CANm bit time and Tq are the values of data bit rate).

This flag is 0 in channel reset mode.

TDCR[6:0] Flags

These bits indicate the transmitter delay compensation result as a multiple of CAN clock frequency (fCAN).

This result depends on the settings of the TDCOC bit and TDCO[6:0] bits in the RSCFDnCFDCmFDCFG register.

This flag is updated at a falling edge between the FDF bit and res bit when the TDCE bit (transmitter delay compensation enable) in the RSCFDnCFDCmFDCFG register is set to 1 and also the TDCOC bit (transmitter delay compensation measurement select) in the RSCFDnCFDCmFDCFG register is set to 0.

This flag is 0 in channel reset mode.

28.4.3.9 RSCFDnCFDCmFDCRC — Channel CANFD CRC Register (m = 0, 1)

Access Size:	RSCFDnCFDCmFDCRC register can be read only in 32-bit units RSCFDnCFDCmFDCRCL, RSCFDnCFDCmFDCRCH registers can be read only in 16-bit units RSCFDnCFDCmFDCRCLL, RSCFDnCFDCmFDCRCLH, RSCFDnCFDCmFDCRCHL, RSCFDnCFDCmFDCRCHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDCmFDCRC: <RSCFDn_base> + H’0510 + (H’20 × m) RSCFDnCFDCmFDCRCL: <RSCFDn_base> + H’0510 + (H’20 × m), RSCFDnCFDCmFDCRCH: <RSCFDn_base> + H’0512 + (H’20 × m) RSCFDnCFDCmFDCRCLL: <RSCFDn_base> + H’0510 + (H’20 × m), RSCFDnCFDCmFDCRCLH: <RSCFDn_base> + H’0511 + (H’20 × m), RSCFDnCFDCmFDCRCHL: <RSCFDn_base> + H’0512 + (H’20 × m), RSCFDnCFDCmFDCRCHH: <RSCFDn_base> + H’0513 + (H’20 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SCNT[3:0]				—	—	—	CRCREG[20:16]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRCREG[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are read as the value after reset.
27 to 24	SCNT[3:0]	All 0	R	Stuff count Indicate a value of the stuff count in a CANFD frame. Bits 25 to 27 indicate the Gray-coded value of the stuff bit count modulo 8 in the transmitted/received frames. Bit 24 indicates an even parity value of bits 25 to 27.
23 to 21	—	All 0	R	Reserved These bits are read as the value after reset.
20 to 0	CRCREG [20:0]	All 0	R	CRC Calculation Data (CRC Length:17 Bits or 21 Bits) These bits show the CRC value calculated based on the transmit message or receive message. When the CRC length is 17 bits, bits b20 to b17 are read as 0.

SCNT[3:0] Flag

When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode enabled), a stuff count bit value of the CANFD frame can be read if a message transmitted/received is a CANFD frame. When the CTME bit is 0 (communication test mode disabled), this flag is always read as 0. This flag is updated at the first bit in the CRC field of the CANFD frame. These bits are cleared to 0 in channel reset mode.

CRCREG[20:0] Flag

When the CTME bit in the RSCFDnCFDCmCTR register is 1 (communication test mode enabled), if transmit or receive message is a CANFD frame (CRC length = 17 or 21 bits), this flag is updated and the CRC value calculated based on the message can be read. When the CRC length of the message is 17 bits, bits b20 to b17 are always read as 0. When a classical CAN frame is transmitted or received, the CRCREG[14:0] value in the RSCFDnCFDCmERFL register is updated. When the CTME bit is 0 (communication test mode disabled), these bits are always read as 0.

28.4.4 Details of Global-related Registers

28.4.4.1 SCFDnCFDGCFG — Global Configuration Register

Access Size:	RSCFDnCFDGCFG register can be read/written in 32-bit units RSCFDnCFDGCFGL, RSCFDnCFDGCFGH registers can be read/written in 16-bit units RSCFDnCFDGCFGLL, RSCFDnCFDGCFGLH, RSCFDnCFDGCFGHL, RSCFDnCFDGCFGHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGCFG: <RSCFDn_base> + H’0084 RSCFDnCFDGCFGL: <RSCFDn_base> + H’0084, RSCFDnCFDGCFGH: <RSCFDn_base> + H’0086 RSCFDnCFDGCFGLL: <RSCFDn_base> + H’0084, RSCFDnCFDGCFGLH: <RSCFDn_base> + H’0085, RSCFDnCFDGCFGHL: <RSCFDn_base> + H’0086, RSCFDnCFDGCFGHH: <RSCFDn_base> + H’0087
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	CMPOC	DCS	MME	DRE	DCE	TPRI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	ITRCP[15:0]	All 0	R/W	Interval Timer Prescaler Set When these bits are set to M, the pclk is divided by M. Setting H’0000 is prohibited when the interval timer is in use.
15 to 13	TSBTCS[2:0]	All 0	R/W	Timestamp Clock Source Select b15 b14 b13 0 0 0: Channel 0 nominal bit time clock 0 0 1: Channel 1 nominal bit time clock 0 1 0: Channel 2 nominal bit time clock 0 1 1: Channel 3 nominal bit time clock 1 0 0: Channel 4 nominal bit time clock 1 0 1: Channel 5 nominal bit time clock 1 1 0: Setting prohibited 1 1 1: Setting prohibited
12	TSSS	0	R/W	Timestamp Source Select 0: pclk/2*1 1: Nominal bit time clock

Bit	Bit Name	Initial Value	R/W	Description
11 to 8	TSP[3:0]	0	R/W	Timestamp Clock Source Division b11 b10 b9 b8 0 0 0 0: Not divided 0 0 0 1: Divided by 2 0 0 1 0: Divided by 4 0 0 1 1: Divided by 8 0 1 0 0: Divided by 16 0 1 0 1: Divided by 32 0 1 1 0: Divided by 64 0 1 1 1: Divided by 128 1 0 0 0: Divided by 256 1 0 0 1: Divided by 512 1 0 1 0: Divided by 1024 1 0 1 1: Divided by 2048 1 1 0 0: Divided by 4096 1 1 0 1: Divided by 8192 1 1 1 0: Divided by 16384 1 1 1 1: Divided by 32768
7, 6	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	CMPOC	0	R/W	Payload Overflow Mode Select 0: No message is stored. 1: Messages are stored and payloads exceeding the buffer size are discarded.
4	DCS	0	R/W	CAN Clock Source Select*[2] 0: clkc 1: clk_xincan
3	MME	0	R/W	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	0	R/W	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	0	R/W	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	0	R/W	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000b.

Note 2. For the CAN clock frequency settings, see **Section 28.1.3, Clock Supply**.

Modify the RSCFDnCFDGCFG register only in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. See **Section 28.8.3.1, Interval Transmission Function**.

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the nominal bit time clock that will be the clock source of the timestamp counter. However, do not select the channel that handles the CANFD frames.

TSSS Bit

This bit is used to select a clock source of the timestamp counter. Select pclk if there is no channel that handles only classical CAN frames.

TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

CMPOC Bit

This bit is used to select operation in case the payload length of received message exceeds the payload storage size of the storage buffer.

When this bit is 0, the received message in which the payload overflows is not stored in the buffer.

When this bit is 1, the received message in which the payload overflows is stored in the buffer. At this time, payloads exceeding the buffer's payload storage size are discarded.

The buffer’s payload storage size is set by the following bits.

- Receive buffer: RMPLS[1:0] bits in the RSCFDnCFDRMNB register
- Receive FIFO buffer: RFPLS[2:0] bits in the RSCFDnCFDRFCCx register
- Transmit/receive FIFO buffer: CFPLS[2:0] bits in the RSCFDnCFDCFCCk register

DCS Bit

When this bit is set to 0, clkc is used as the clock source of the CAN clock (fCAN).

When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN).

For the CAN clock frequency settings, see **Table 28.6, Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI**.

MME Bit

Setting this bit to 1 makes the mirror function available.

DRE Bit

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of H’00 is stored in each data byte beyond the DLC value of the receive rule.

The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

DCE Bit

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCFDnCFDGAFLP0_j register to 0000b before clearing the DCE bit in the RSCFDnCFDGCFG register to 0.

TPRI Bit

This bit is used to set the transmit priority.

When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.

While the transmit queue is in use, this bit should be set to 0.

28.4.4.2 RSCFDnCFDGCTR — Global Control Register

Access Size: RSCFDnCFDGCTR register can be read/written in 32-bit units
RSCFDnCFDGCTRL, RSCFDnCFDGCTRH registers can be read/written in 16-bit units
RSCFDnCFDGCTRLL, RSCFDnCFDGCTRLH, RSCFDnCFDGCTRHL, RSCFDnCFDGCTRHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDGCTR: <RSCFDn_base> + H'0088
RSCFDnCFDGCTRL: <RSCFDn_base> + H'0088, RSCFDnCFDGCTRH: <RSCFDn_base> + H'008A
RSCFDnCFDGCTRLL: <RSCFDn_base> + H'0088, RSCFDnCFDGCTRLH: <RSCFDn_base> + H'0089,
RSCFDnCFDGCTRHL: <RSCFDn_base> + H'008A, RSCFDnCFDGCTRHH: <RSCFDn_base> + H'008B

Initial Value: H'0000 0005

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CMPOFIE	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
16	TSRST	0	R/W	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 12	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
11	CMPOFIE	0	R/W	Payload Overflow Interrupt Enable 0: A payload overflow interrupt is disabled. 1: A payload overflow interrupt is enabled.
10	THLEIE	0	R/W	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	0	R/W	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	0	R/W	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	GSLPR	1	R/W	Global Stop Mode 0: Other than global stop mode 1: Global stop mode

Bit	Bit Name	Initial Value	R/W	Description
1, 0	GMDC[1:0]	01b	R/W	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCFDnCFDGTSC register is cleared to H’0000.

CMPOFIE Bit

When the CMPOF flag in the RSCFDnCFDGERFL register is set to 1 after the CMPOFIE bit is set to 1, an interrupt request occurs. Modify this bit only in global reset mode.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

GSLPR Bit

Setting this bit to 1 places the RSCAN module into global stop mode.

Clearing this bit to 0 makes the RSCAN module leave from global stop mode.

This bit should not be modified in global operating mode or global test mode.

GMDC[1:0] Bits

These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see **Section 28.6.1, Global Modes**. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.

28.4.4.3 RSCFDnCFDGSTS — Global Status Register

Access Size:	RSCFDnCFDGSTS register can be read only in 32-bit units RSCFDnCFDGSTSL, RSCFDnCFDGSTSH registers can be read only in 16-bit units RSCFDnCFDGSTSLL, RSCFDnCFDGSTSLH, RSCFDnCFDGSTSHL, RSCFDnCFDGSTSHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDGSTS: <RSCFDn_base> + H'008C RSCFDnCFDGSTSL: <RSCFDn_base> + H'008C, RSCFDnCFDGSTSH: <RSCFDn_base> + H'008E RSCFDnCFDGSTSLL: <RSCFDn_base> + H'008C, RSCFDnCFDGSTSLH: <RSCFDn_base> + H'008D, RSCFDnCFDGSTSHL: <RSCFDn_base> + H'008E, RSCFDnCFDGSTSHH: <RSCFDn_base> + H'008F
Initial Value:	H'0000 000D

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are read as the value after reset.
3	GRAMINIT	1	R	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	1	R	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	0	R	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	1	R	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.

This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to global stop mode, and is cleared to 0 when the CAN module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global test mode, and is cleared to 0 when the CAN module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global reset mode, and is cleared to 0 when the CAN module has exited global reset mode. This flag remains 1 even when the CAN module has transitioned from global reset mode to global stop mode.

28.4.4.4 RSCFDnCFDGERFL — Global Error Flag Register

Access Size:	RSCFDnCFDGERFL register can be read/written in 32-bit units RSCFDnCFDGERFLL, RSCFDnCFDGERFLH registers can be read/written in 16-bit units RSCFDnCFDGERFLLL, RSCFDnCFDGERFLLH, RSCFDnCFDGERFLHL, RSCFDnCFDGERFLHH registers can be read/written in 8- bit units
Address(es):	RSCFDnCFDGERFL: <RSCFDn_base> + H’0090 RSCFDnCFDGERFLL: <RSCFDn_base> + H’0090, RSCFDnCFDGERFLH: <RSCFDn_base> + H’0092 RSCFDnCFDGERFLLL: <RSCFDn_base> + H’0090, RSCFDnCFDGERFLLH: <RSCFDn_base> + H’0091, RSCFDnCFDGERFLHL: <RSCFDn_base> + H’0092, RSCFDnCFDGERFLHH: <RSCFDn_base> + H’0093
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CMPOF	THLES	MES	DEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R	R	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
31 to 14, 7, 6, 4	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
13 to 8, 5	—	All 0	R	Reserved When read, an undefined value is returned.The write value should be the value after reset.
3	CMPOF	1	R/W*1	Payload Overflow Flag 0: No payload overflow has occurred. 1: A payload overflow has occurred.
2	THLES	1	R	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	0	R	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	1	R/W*1	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

All flags in the RSCFDnCFDGERFL register are cleared to 0 in global reset mode.

CMPOF Flag

When a payload overflow occurs in any of channel m (m = 0, 1), the CMPOF flag is set to 1. This flag can be cleared to 0 by writing 0 to this bit by the program.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCFDnCFDTHLSTSm register (m = 0, 1) is set to 1.

This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCFDnCFDRFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCFDnCFDCFSTSk register (k = 0 to 5) is set to 1.

This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

Note

To clear the flag of this register to 0, use a store instruction to write 0 to the given flag and 1 to the other flags.

28.4.4.5 RSCFDnCFDGTSC — Global Timestamp Counter Register

Access Size: RSCFDnCFDGTSC register can be read only in 32-bit units
RSCFDnCFDGTSCL, RSCFDnCFDGTSCH registers can be read only in 16-bit units

Address(es): RSCFDnCFDGTSC: <RSCFDn_base> + H’0094
RSCFDnCFDGTSCL: <RSCFDn_base> + H’0094, RSCFDnCFDGTSCH: <RSCFDn_base> + H’0096

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset.
15 to 0	TS[15:0]	All 0	R	Timestamp Value The timestamp counter value can be read. Counter Value: H’0000 to H’FFFF

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. Furthermore, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.

The timestamp counter starts and stops counting differently, depending on the count source.

- When the TSSS bit in the RSCFDnCFDGCFG register is 0 (pclk):
  The timestamp counter starts counting when the RSCAN module has transitioned to global operating mode. This counter stops counting when the RSCAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm nominal bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.
  This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

28.4.4.6 RSCFDnCFDGTINTSTS0 — Global TX Interrupt Status Register 0

Access Size:	RSCFDnCFDGTINTSTS0 register can be read only in 32-bit units RSCFDnCFDGTINTSTS0L, RSCFDnCFDGTINTSTS0H registers can be read only in 16-bit units RSCFDnCFDGTINTSTS0LL, RSCFDnCFDGTINTSTS0LH, RSCFDnCFDGTINTSTS0HL, RSCFDnCFDGTINTSTS0HH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDGTINTSTS0: <RSCFDn_base> + H’0460 RSCFDnCFDGTINTSTS0L: <RSCFDn_base> + H’0460, RSCFDnCFDGTINTSTS0H: <RSCFDn_base> + H’0462 RSCFDnCFDGTINTSTS0LL: <RSCFDn_base> + H’0460, RSCFDnCFDGTINTSTS0LH: <RSCFDn_base> + H’0461, RSCFDnCFDGTINTSTS0HL: <RSCFDn_base> + H’0462, RSCFDnCFDGTINTSTS0HH: <RSCFDn_base> + H’0463
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are read as the value after reset.
12	THIF1	0	R*1	Channel 1 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	0	R*1	Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
10	TQIF1	0	R*1	Channel 1 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	0	R*1	Channel 1 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
8	TSIF1	0	R*1	Channel 1 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset.
4	THIF0	0	R*1	Channel 0 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	0	R*1	Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.

Bit	Bit Name	Initial Value	R/W	Description
2	TQIF0	0	R*[1]	Channel 0 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF0	0	R*[1]	Channel 0 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
0	TSIF0	0	R*[1]	Channel 0 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

TSIFm Bits

The TSIFm bit is set to 1 when the TMIEp bit in the RSCFDnCFDTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to 10b (transmit completed without abort request) or 11b (transmit completed with abort request).

When the TMTRF[1:0] flags are cleared to 00b under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.

TAIFm Bits

The TAIFm bit is set to 1 when the TAIE bit in the RSCFDnCFDCmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to 01b (transmit abort completed).

This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00b after the transmit abort is completed.

TQIFm Bits

When the TXQIE bit in the RSCFDnCFDTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCFDnCFDTXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.

When the TXQIF bit (transmit queue interrupt request) in the RSCFDnCFDTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

CFTIFm Bits

When the CFTXIE bit in the RSCFDnCFDCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCFDnCFDCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.

When the CFTXIF bit is cleared to 0 under the conditions that the CFTIFm bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

THIFm Bits

When the THLIE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCFDnCFDTHLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.

When the THLIF bit in the RSCFDnCFDTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

28.4.4.7 RSCFDnCFDGFDCFG — Global FD Configuration Register

Access Size:	RSCFDnCFDGFDCFG register can be read/written in 32-bit units RSCFDnCFDGFDCFG L, RSCFDnCFDGFDCFG H registers can be read/written in 16-bit units RSCFDnCFDGFDCFG LL, RSCFDnCFDGFDCFG LH, RSCFDnCFDGFDCFG HL, RSCFDnCFDGFDCFG HH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGFDCFG: <RSCFDn_base> + H’0474 RSCFDnCFDGFDCFG L: <RSCFDn_base> + H’0474, RSCFDnCFDGFDCFG H: <RSCFDn_base> + H’0476 RSCFDnCFDGFDCFG LL: <RSCFDn_base> + H’0474, RSCFDnCFDGFDCFG LH: <RSCFDn_base> + H’0475, RSCFDnCFDGFDCFG HL: <RSCFDn_base> + H’0476, RSCFDnCFDGFDCFG HH: <RSCFDn_base> + H’0477
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSCCFG[1:0]		—	—	—	—	—	—	—	PRED
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R/W	Reserved These bits are read as the value after reset. The write value should be the value after reset.
9, 8	TSCCFG[1:0]	All 0	R/W	Time-stamp capture setting bit b9 b8 0 0: Captured at a sample point in the SOF bit. 0 1: Captured when a valid frame has been transmitted/received. 1 0: Captured at a sample point of the res bit.*1 1 1: Setting prohibited.
7 to 1	—	All 0	R/W	Reserved These bits are read as the value after reset. The write value should be the value after reset.
0	PRED	0	R/W	Protocol exception event detection disabled bit 0: Protocol exception event detection enabled 1: Protocol exception event detection disabled

Note 1. When a Classical CAN frame is transmitted/received, a time-stamp value will be captured at the sample point in the SOF bit.

TSCCFG[1:0] Bits

Select a point where a time-stamp value is captured. Modify this bit only in global reset mode.

PRED Bit

Setting this bit to 1 disables the protocol exception event detection. When a protocol exception event is detected while this bit is set to 1, the event is regarded as a form error and an error frame is output. Modify this bit only in global reset mode.

28.4.5 Details of Receive Rule-related Registers

28.4.5.1 RSCFDnCFDGAFLECTR — Receive Rule Entry Control Register

Access Size:	RSCFDnCFDGAFLECTR register can be read/written in 32-bit units RSCFDnCFDGAFLECTRL, RSCFDnCFDGAFLECTRH registers can be read/written in 16-bit units RSCFDnCFDGAFLECTRLL, RSCFDnCFDGAFLECTRLH, RSCFDnCFDGAFLECTRHL, RSCFDnCFDGAFLECTRHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGAFLECTR: <RSCFDn_base> + H’0098 RSCFDnCFDGAFLECTRL: <RSCFDn_base> + H’0098, RSCFDnCFDGAFLECTRH: <RSCFDn_base> + H’009A RSCFDnCFDGAFLECTRLL: <RSCFDn_base> + H’0098, RSCFDnCFDGAFLECTRLH: <RSCFDn_base> + H’0099, RSCFDnCFDGAFLECTRHL: <RSCFDn_base> + H’009A, RSCFDnCFDGAFLECTRHH: <RSCFDn_base> + H’009B
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
8	AFLDAE	0	R/W	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
4 to 0	AFLPN[4:0]	All 0	R/W	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000b) to page 7 (00111b).

AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.

Set the AFLDAE bit to 1 only in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page. Set these bits to a value within the range of 00000b to 00111b.

28.4.5.2 RSCFDnCFDGAFLCFG0 — Receive Rule Configuration Register 0

Access Size:	RSCFDnCFDGAFLCFG0 register can be read/written in 32-bit units RSCFDnCFDGAFLCFG0L, RSCFDnCFDGAFLCFG0H registers can be read/written in 16-bit units RSCFDnCFDGAFLCFG0LL, RSCFDnCFDGAFLCFG0LH, RSCFDnCFDGAFLCFG0HL, RSCFDnCFDGAFLCFG0HH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGAFLCFG0: <RSCFDn_base> + H’009C RSCFDnCFDGAFLCFG0L: <RSCFDn_base> + H’009C, RSCFDnCFDGAFLCFG0H: <RSCFDn_base> + H’009E RSCFDnCFDGAFLCFG0LL: <RSCFDn_base> + H’009C, RSCFDnCFDGAFLCFG0LH: <RSCFDn_base> + H’009D, RSCFDnCFDGAFLCFG0HL: <RSCFDn_base> + H’009E, RSCFDnCFDGAFLCFG0HH: <RSCFDn_base> + H’009F
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RNC0[7:0]	All 0	R/W	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	All 0	R/W	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 0	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCFDnCFDGAFLCFG0 register only in global reset mode.

Up to 64 × (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table. Set these bits to a value within the range of H’00 to H’80.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table. Set these bits to a value within the range of H’00 to H’80.

28.4.5.3 RSCFDnCFDGAFLIDj — Receive Rule ID Register (j = 0 to 15)

Access Size:	RSCFDnCFDGAFLIDj register can be read/written in 32-bit units RSCFDnCFDGAFLIDjL, RSCFDnCFDGAFLIDjH registers can be read/written in 16-bit units RSCFDnCFDGAFLIDjLL, RSCFDnCFDGAFLIDjLH, RSCFDnCFDGAFLIDjHL, RSCFDnCFDGAFLIDjHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGAFLIDj: <RSCFDn_base> + H’1000 + (H’10 × j) RSCFDnCFDGAFLIDjL: <RSCFDn_base> + H’1000 + (H’10 × j), RSCFDnCFDGAFLIDjH: <RSCFDn_base> + H’1002 + (H’10 × j) RSCFDnCFDGAFLIDjLL: <RSCFDn_base> + H’1000 + (H’10 × j), RSCFDnCFDGAFLIDjLH: <RSCFDn_base> + H’1001 + (H’10 × j), RSCFDnCFDGAFLIDjHL: <RSCFDn_base> + H’1002 + (H’10 × j), RSCFDnCFDGAFLIDjHH: <RSCFDn_base> + H’1003 + (H’10 × j)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDE	0	R/W	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	0	R/W	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	0	R/W	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
28 to 0	GAFLID[28:0]	All 0	R/W	ID Set Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCFDnCFDGAFLIDj register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.

When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

28.4.5.4 RSCFDnCFDGAFLMj — Receive Rule Mask Register (j = 0 to 15)

Access Size:	RSCFDnCFDGAFLMj register can be read/written in 32-bit units RSCFDnCFDGAFLMjL, RSCFDnCFDGAFLMjH registers can be read/written in 16-bit units RSCFDnCFDGAFLMjLL, RSCFDnCFDGAFLMjLH, RSCFDnCFDGAFLMjHL, RSCFDnCFDGAFLMjHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGAFLMj: <RSCFDn_base> + H’1004 + (H’10 × j) RSCFDnCFDGAFLMjL: <RSCFDn_base> + H’1004 + (H’10 × j), RSCFDnCFDGAFLMjH: <RSCFDn_base> + H’1006 + (H’10 × j) RSCFDnCFDGAFLMjLL: <RSCFDn_base> + H’1004 + (H’10 × j), RSCFDnCFDGAFLMjLH: <RSCFDn_base> + H’1005 + (H’10 × j), RSCFDnCFDGAFLMjHL: <RSCFDn_base> + H’1006 + (H’10 × j), RSCFDnCFDGAFLMjHH: <RSCFDn_base> + H’1007 + (H’10 × j)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDEM	GAFLRTRM	—	GAFLIDM[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDEM	0	R/W	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	0	R/W	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared
29	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
28 to 0	GAFLIDM [28:0]	All 0	R/W	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCFDnCFDGAFLMj register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCFDnCFDGAFLIDj register.

When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

28.4.5.5 RSCFDnCFDGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access Size: RSCFDnCFDGAFLP0_j register can be read/written in 32-bit units

RSCFDnCFDGAFLP0_jL, RSCFDnCFDGAFLP0_jH registers can be read/written in 16-bit units

RSCFDnCFDGAFLP0jLL, RSCFDnCFDGAFLP0jLH, RSCFDnCFDGAFLP0jHL, RSCFDnCFDGAFLP0jHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDGAFLP0_j: <RSCFDn_base> + H’1008 + (H’10 × j)

RSCFDnCFDGAFLP0_jL: <RSCFDn_base> + H’1008 + (H’10 × j),

RSCFDnCFDGAFLP0_jH: <RSCFDn_base> + H’100A + (H’10 × j)

RSCFDnCFDGAFLP0_jLL: <RSCFDn_base> + H’1008 + (H’10 × j),

RSCFDnCFDGAFLP0_jLH: <RSCFDn_base> + H’1009 + (H’10 × j),

RSCFDnCFDGAFLP0_jHL: <RSCFDn_base> + H’100A + (H’10 × j),

RSCFDnCFDGAFLP0_jHH: <RSCFDn_base> + H’100B + (H’10 × j)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLRMV	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GAFLDLC [3:0]	All 0	R/W	Receive Rule DLC

b31	b30	b29	b28	Classical CAN Frame	CANFD Frame
0	0	0	0	0 data bytes	
0	0	0	1	1 data byte	
0	0	1	0	2 data bytes	
0	0	1	1	3 data bytes	
0	1	0	0	4 data bytes	
0	1	0	1	5 data bytes	
0	1	1	0	6 data bytes	
0	1	1	1	7 data bytes	
1	0	0	0	8 data bytes	
1	0	0	1	8 data bytes	12 data bytes
1	0	1	0		16 data bytes
1	0	1	1		20 data bytes
1	1	0	0		24 data bytes
1	1	0	1		32 data bytes
1	1	1	0		48 data bytes
1	1	1	1		64 data bytes

Bit	Bit Name	Initial Value	R/W	Description
27 to 16	GAFLPTR [11:0]	All 0	R/W	Receive Rule Label Set the 12-bit label information.
15	GAFLRMV	0	R/W	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP [6:0]	All 0	R/W	Receive Buffer Number Select Set the receive buffer number to store receive messages.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Modify the RSCFDnCFDGAFLP0_j register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check.

Setting these bits to 0000b disables the DLC check function allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCFDnCFDRMNB register.

28.4.5.6 RSCFDnCFDGAFLP1_j — Receive Rule Pointer 1 Register (j = 0 to 15)

Access Size: RSCFDnCFDGAFLP1_j register can be read/written in 32-bit units
RSCFDnCFDGAFLP1_jL, RSCFDnCFDGAFLP1_jH registers can be read/written in 16-bit units
RSCFDnCFDGAFLP1_jLL, RSCFDnCFDGAFLP1_jLH, RSCFDnCFDGAFLP1_jHL, RSCFDnCFDGAFLP1_jHH registers can be read/ written in 8-bit units

Address(es): RSCFDnCFDGAFLP1_j: <RSCFDn_base> + H’100C$_H$ + (H’10 × j)
RSCFDnCFDGAFLP1_jL: <RSCFDn_base> + H’100C$_H$ + (H’10 × j),
RSCFDnCFDGAFLP1_jH: <RSCFDn_base> + H’100E$_H$ + (H’10 × j)
RSCFDnCFDGAFLP1_jLL: <RSCFDn_base> + H’100C$_H$ + (H’10 × j),
RSCFDnCFDGAFLP1_jLH: <RSCFDn_base> + H’100D$_H$ + (H’10 × j),
RSCFDnCFDGAFLP1_jHL: <RSCFDn_base> + H’100E$_H$ + (H’10 × j),
RSCFDnCFDGAFLP1_jHH: <RSCFDn_base> + H’100F$_H$ + (H’10 × j)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	GAFLFDP[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
13 to 8	GAFLFDP [13:8]	All 0	R/W	Transmit/Receive FIFO Buffer k Select (Bit position –8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected.
7 to 0	GAFLFDP [7:0]	All 0	R/W	Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected.

Modify the RSCFDnCFDGAFLP1_j register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP[13:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCFDnCFDGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCFDnCFDCFCCk register are set to 00b (receive mode) or 10b (gateway mode) are selectable.

28.4.6 Details of Receive Buffer-related Registers

28.4.6.1 RSCFDnCFDRMNB — Receive Buffer Number Register

Access Size:	RSCFDnCFDRMNB register can be read/written in 32-bit units RSCFDnCFDRMNBL, RSCFDnCFDRMNBH registers can be read/written in 16-bit units RSCFDnCFDRMNBLL, RSCFDnCFDRMNBLH, RSCFDnCFDRMNBHL, RSCFDnCFDRMNBHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDRMNB: <RSCFDn_base> + H’00A4 RSCFDnCFDRMNBL: <RSCFDn_base> + H’00A4, RSCFDnCFDRMNBH: <RSCFDn_base> + H’00A6 RSCFDnCFDRMNBLL: <RSCFDn_base> + H’00A4, RSCFDnCFDRMNBLH: <RSCFDn_base> + H’00A5, RSCFDnCFDRMNBHL: <RSCFDn_base> + H’00A6, RSCFDnCFDRMNBHH: <RSCFDn_base> + H’00A7
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMPLS[1:0]		NRXMB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
9, 8	RMPLS[1:0]	All 0	R/W	Receive Buffer Payload Storage Size Select b9 b8 0 0: 8 bytes 0 1: 12 bytes 1 0: 16 bytes 1 1: 20 bytes
7 to 0	NRXMB[7:0]	All 0	R/W	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 32.

Modify the RSCFDnCFDRMNB register only in global reset mode.

RMPLS[1:0] Bits

These bits are used to select the maximum payload size that can be stored in the receive buffer.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels).

Setting these bits all to 0 makes receive buffers unavailable.

28.4.6.2 RSCFDnCFDRMNDy — Receive Buffer New Data Register (y = 0)

Access Size: RSCFDnCFDRMNDy register can be read/written in 32-bit units

RSCFDnCFDRMNDyL, RSCFDnCFDRMNDyH registers can be read/written in 16-bit units

RSCFDnCFDRMNDyLL, RSCFDnCFDRMNDyLH, RSCFDnCFDRMNDyHL, RSCFDnCFDRMNDyHH registers can be read/written in 8- bit units

Address(es): RSCFDnCFDRMNDy: <RSCFDn_base> + H’00A8 + (H’04 × y)

RSCFDnCFDRMNDyL: <RSCFDn_base> + H’00A8 + (H’04 × y),

RSCFDnCFDRMNDyH: <RSCFDn_base> + H’00AA + (H’04 × y)

RSCFDnCFDRMNDyLL: <RSCFDn_base> + H’00A8 + (H’04 × y),

RSCFDnCFDRMNDyLH: <RSCFDn_base> + H’00A9 + (H’04 × y),

RSCFDnCFDRMNDyHL: <RSCFDn_base> + H’00AA + (H’04 × y),

RSCFDnCFDRMNDyHH: <RSCFDn_base> + H’00AB + (H’04 × y)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	RMNSq	All 0	R/W	Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	All 0	R/W	Receive Buffer Receive Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCFDnCFDRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 31)

Each RMNSq flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write “0” to the flag and “1” to other flags. These bits cannot be set to 0 while a message is being stored. The message storing time depends on the storage payload size of the receive buffer. When the RMPLS[1:0] value in the RSCFDnCFDRMNB register is 00b (8 bytes), the message storing time is 12 pclk clock cycles. When the RMPLS[1:0] value is 11b (20 bytes), the message storing time is 18 pclk clock cycles. (2 pclk clock cycles per 4 bytes of storage payload size).

These flags are cleared to 0 in global reset mode.

28.4.6.3 RSCFDnCFDRMIDq — Receive Buffer ID Register (q = 0 to 31)

Access Size:	RSCFDnCFDRMIDq register can be read only in 32-bit units RSCFDnCFDRMIDqL, RSCFDnCFDRMIDqH registers can be read only in 16-bit units RSCFDnCFDRMIDqLL, RSCFDnCFDRMIDqLH, RSCFDnCFDRMIDqHL, RSCFDnCFDRMIDqHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDRMIDq: <RSCFDn_base> + H’2000 + (H’20 × q) RSCFDnCFDRMIDqL: <RSCFDn_base> + H’2000 + (H’20 × q), RSCFDnCFDRMIDqH: <RSCFDn_base> + H’2002 + (H’20 × q) RSCFDnCFDRMIDqLL: <RSCFDn_base> + H’2000 + (H’20 × q), RSCFDnCFDRMIDqLH: <RSCFDn_base> + H’2001 + (H’20 × q), RSCFDnCFDRMIDqHL: <RSCFDn_base> + H’2002 + (H’20 × q), RSCFDnCFDRMIDqHH: <RSCFDn_base> + H’2003 + (H’20 × q)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RMIDE	0	R	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	0	R	Receive Buffer RTR/RRS ● When the received message is a classical CAN frame 0: Data frame 1: Remote frame ● When the received message is a CANFD frame The RRS bit value of the received message can be read.
29	—	0	R	Reserved This bit is read as the value after reset.
28 to 0	RMID[28:0]	All 0	R	Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer.

RMRTR Bit

When the received message is a classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer. When the received message is a CANFD frame, this bit indicates the RRS bit value in the message.

RMID[28:0] Bits

These bits contain the ID of the message stored in the receive buffer.

28.4.6.4 RSCFDnCFDRMPTRq — Receive Buffer Pointer Register (q = 0 to 31)

Access Size:	RSCFDnCFDRMPTRq register can be read only in 32-bit units RSCFDnCFDRMPTRqL, RSCFDnCFDRMPTRqH registers can be read only in 16-bit units RSCFDnCFDRMPTRqLL, RSCFDnCFDRMPTRqLH, RSCFDnCFDRMPTRqHL, RSCFDnCFDRMPTRqHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDRMPTRq: <RSCFDn_base> + H’2004 + (H’20 × q) RSCFDnCFDRMPTRqL: <RSCFDn_base> + H’2004 + (H’20 × q), RSCFDnCFDRMPTRqH: <RSCFDn_base> + H’2006 + (H’20 × q) RSCFDnCFDRMPTRqLL: <RSCFDn_base> + H’2004 + (H’20 × q), RSCFDnCFDRMPTRqLH: <RSCFDn_base> + H’2005 + (H’20 × q), RSCFDnCFDRMPTRqHL: <RSCFDn_base> + H’2006 + (H’20 × q), RSCFDnCFDRMPTRqHH: <RSCFDn_base> + H’2007 + (H’20 × q)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31 to 28</td><td>RMDLC[3:0]</td><td>All 0</td><td>R</td><td>Receive Buffer DLC Data
<table>
<tr><th>b31</th><th>b30</th><th>b29</th><th>b28</th><th>Classical CAN Frame</th><th>CANFD Frame</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0 data bytes</td><td></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1 data byte</td><td></td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>2 data bytes</td><td></td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>3 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>4 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>5 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>6 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>7 data bytes</td><td></td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>8 data bytes</td><td></td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>8 data bytes</td><td>12 data bytes</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td></td><td>16 data bytes</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td></td><td>20 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td></td><td>24 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td></td><td>32 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td></td><td>48 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td></td><td>64 data bytes</td></tr>
</table></td></tr>
<tr><td>27 to 16</td><td>RMPTR[11:0]</td><td>All 0</td><td>R</td><td>Receive Buffer Label Data
Label information of the received message.</td></tr>
<tr><td>15 to 0</td><td>RMTS[15:0]</td><td>All 0</td><td>R</td><td>Receive Buffer Timestamp Data
Timestamp value of the received message.</td></tr>
</table>

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer. The number of bytes of the payload to be stored in the receive buffer is determined by the RMPLS[1:0] bits in the RSCFDnCFDRMNB register.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

28.4.6.5 RSCFDnCFDRMFDSTSq — Receive Buffer CANFD Status Register (q = 0 to 31)

Access Size: RSCFDnCFDRMFDSTSq register can be read only in 32-bit units
RSCFDnCFDRMFDSTSqL, RSCFDnCFDRMFDSTSqH registers can be read only in 16-bit units
RSCFDnCFDRMFDSTSqLL, RSCFDnCFDRMFDSTSqLH, RSCFDnCFDRMFDSTSqHL, RSCFDnCFDRMFDSTSqHH registers can be read only in 8-bit units

Address(es): RSCFDnCFDRMFDSTSq: <RSCFDn_base> + H'2008 + (H'20 × q)
RSCFDnCFDRMFDSTSqL: <RSCFDn_base> + H'2008 + (H'20 × q),
RSCFDnCFDRMFDSTSqH: <RSCFDn_base> + H'200A + (H'20 × q)
RSCFDnCFDRMFDSTSqLL: <RSCFDn_base> + H'2008 + (H'20 × q),
RSCFDnCFDRMFDSTSqLH: <RSCFDn_base> + H'2009 + (H'20 × q),
RSCFDnCFDRMFDSTSqHL: <RSCFDn_base> + H'200A + (H'20 × q),
RSCFDnCFDRMFDSTSqHH: <RSCFDn_base> + H'200B + (H'20 × q)

Initial Value: H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RMFDF	RMBRS	RMESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as the value after reset.
2	RMFDF	0	R	FDF 0: Classical CAN frame 1: CANFD frame
1	RMBRS	0	R	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	RMESI	0	R	ESI 0: Error active node 1: Error passive node

RMFDF Bit

This bit indicates the FD format (classical CAN frame or CANFD frame) of the message stored in the receive buffer.

RMBRS Bit

When the RMFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.

RMESI Bit

When the RMFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.

28.4.6.6 RSCFDnCFDRMDFb_q — Receive Buffer Data Field b Register (b = 0 to 4, q = 0 to 31)

Access Size: RSCFDnCFDRMDFb_q register can be read only in 32-bit units

RSCFDnCFDRMDFb_qL, RSCFDnCFDRMDFb_qH registers can be read only in 16-bit units

RSCFDnCFDRMDFb_qLL, RSCFDnCFDRMDFb_qLH, RSCFDnCFDRMDFb_qHL, RSCFDnCFDRMDFb_qHH registers can be read only in 8-bit units

Address(es): RSCFDnCFDRMDFb_q: <RSCFDn_base> + H’200C + (H’04 × b) + (H’20 × q)

RSCFDnCFDRMDFb_qL: <RSCFDn_base> + H’200C + (H’04 × b) + (H’20 × q),

RSCFDnCFDRMDFb_qH: <RSCFDn_base> + H’200E + (H’04 × b) + (H’20 × q)

RSCFDnCFDRMDFb_qLL: <RSCFDn_base> + H’200C + (H’04 × b) + (H’20 × q),

RSCFDnCFDRMDFb_qLH: <RSCFDn_base> + H’200D + (H’04 × b) + (H’20 × q),

RSCFDnCFDRMDFb_qHL: <RSCFDn_base> + H’200E + (H’04 × b) + (H’20 × q),

RSCFDnCFDRMDFb_qHH: <RSCFDn_base> + H’200F + (H’04 × b) + (H’20 × q)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB4 × b + 3 [7:0]								RMDB4 × b + 2 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB4 × b + 1 [7:0]								RMDB4 × b + 0 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RMDB4 × b + 3 [7:0]	All 0	R	Receive Buffer Data Byte 4 × b + 3 Receive Buffer Data Byte 4 × b + 2 Receive Buffer Data Byte 4 × b + 1 Receive Buffer Data Byte 4 × b + 0 Data for a message stored in the receive buffer can be read.
23 to 16	RMDB4 × b + 2 [7:0]	All 0	R	
15 to 8	RMDB4 × b + 1 [7:0]	All 0	R	
7 to 0	RMDB4 × b + 0 [7:0]	All 0	R	

When the RMDLC[3:0] value in the RSCFDnCFDRMPTRq register is smaller than the payload storage size of the receive buffer, data bytes for which no data is set are read as H’00.

Specify the payload storage size of the receive buffer by the RMPLS[1:0] bits in the RSCFDnCFDRMNB register. Do not read or write the RSCFDnCFDRMDFb_q register corresponding to an area larger than the specified size.

28.4.7 Details of Receive FIFO Buffer-related Registers

28.4.7.1 RSCFDnCFDRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

Access Size:	RSCFDnCFDRFCCx register can be read/written in 32-bit units RSCFDnCFDRFCCxL, RSCFDnCFDRFCCxH registers can be read/written in 16-bit units RSCFDnCFDRFCCxLL, RSCFDnCFDRFCCxLH, RSCFDnCFDRFCCxHL, RSCFDnCFDRFCCxHH registers can be read/written in 8- bit units
Address(es):	RSCFDnCFDRFCCx: <RSCFDn_base> + H’00B8 + (H’04 × x) RSCFDnCFDRFCCxL: <RSCFDn_base> + H’00B8 + (H’04 × x), RSCFDnCFDRFCCxH: <RSCFDn_base> + H’00BA + (H’04 × x) RSCFDnCFDRFCCxLL: <RSCFDn_base> + H’00B8 + (H’04 × x), RSCFDnCFDRFCCxLH: <RSCFDn_base> + H’00B9 + (H’04 × x), RSCFDnCFDRFCCxHL: <RSCFDn_base> + H’00BA + (H’04 × x), RSCFDnCFDRFCCxHH: <RSCFDn_base> + H’00BB + (H’04 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	RFPLS[2:0]			—	—	RFIE	RFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
15 to 13	RFIGCV[2:0]	All 0	R/W	Receive FIFO Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	0	R/W	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	RFDC[2:0]	All 0	R/W	Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
6 to 4	RFPLS[2:0]	All 0	R/W	Receive FIFO Buffer Payload Storage Size Select b6 b5 b4 0 0 0: 8 bytes 0 0 1: 12 bytes 0 1 0: 16 bytes 0 1 1: 20 bytes 1 0 0: 24 bytes 1 0 1: 32 bytes 1 1 0: 48 bytes 1 1 1: 64 bytes
3, 2	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
1	RFIE	0	R/W	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.
0	RFE	0	R/W	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]).

When the RFDC[2:0] bits are set to 001b (4 messages), set the RFIGCV[2:0] bits to 001b, 011b, 101b, or 111b. Modify these bits only in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000b, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFPLS[2:0] Bits

These bits are used to select the maximum payload size that can be stored in the receive FIFO buffer. Modify these bits only in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCFDnCFDRFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode. Set this bit to 1 with another instruction after the settings to all bits in the RSCFDnCFDRFCCx register have been done. This bit is cleared to 0 in global reset mode.

28.4.7.2 RSCFDnCFDRFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7)

Access Size:	RSCFDnCFDRFSTSx register can be read/written in 32-bit units RSCFDnCFDRFSTSxL, RSCFDnCFDRFSTSxH registers can be read/written in 16-bit units RSCFDnCFDRFSTSxLL, RSCFDnCFDRFSTSxLH, RSCFDnCFDRFSTSxHL, RSCFDnCFDRFSTSxHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDRFSTSx: <RSCFDn_base> + H’00D8 + (H’04 × x) RSCFDnCFDRFSTSxL: <RSCFDn_base> + H’00D8 + (H’04 × x), RSCFDnCFDRFSTSxH: <RSCFDn_base> + H’00DA + (H’04 × x) RSCFDnCFDRFSTSxLL: <RSCFDn_base> + H’00D8 + (H’04 × x), RSCFDnCFDRFSTSxLH: <RSCFDn_base> + H’00D9 + (H’04 × x), RSCFDnCFDRFSTSxHL: <RSCFDn_base> + H’00DA + (H’04 × x), RSCFDnCFDRFSTSxHH: <RSCFDn_base> + H’00DB + (H’04 × x)
Initial Value:	H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*[1]	R/W*[1]	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	RFMC[7:0]	All 0	R	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	—	All 0	R	Reserved These bits are read as the value after reset. When writing to these bits, write the value after reset.
3	RFIF	0	R/W*[1]	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	0	R/W*[1]	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	0	R	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	1	R	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

RFMC[7:0] Flag

This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes H’00 when the RFE bit in the RSCFDnCFDRFCCx register is set to 0.

This flag is H’00 in global reset mode.

RFIF Flag

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCFDnCFDRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFMLT Flag

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.

This flag is cleared to 0 in global reset mode or by writing 0 to this flag.

Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCFDnCFDRFCCx register.

If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCFDnCFDRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCFDnCFDRFCCx register is 0 or in global reset mode.

This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

NOTE

To clear the RFMLT or RFIF flag to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

28.4.7.3 RSCFDnCFDRFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access Size:	RSCFDnCFDRFPCTRx register can only be written in 32-bit units RSCFDnCFDRFPCTRxL, RSCFDnCFDRFPCTRxH registers can only be written in 16-bit units RSCFDnCFDRFPCTRxLL, RSCFDnCFDRFPCTRxLH, RSCFDnCFDRFPCTRxHL, RSCFDnCFDRFPCTRxHH registers can only be written in 8-bit units
Address(es):	RSCFDnCFDRFPCTRx: <RSCFDn_base> + H’00F8 + (H’04 × x) RSCFDnCFDRFPCTRxL: <RSCFDn_base> + H’00F8 + (H’04 × x), RSCFDnCFDRFPCTRxH: <RSCFDn_base> + H’00FA + (H’04 × x) RSCFDnCFDRFPCTRxLL: <RSCFDn_base> + H’00F8 + (H’04 × x), RSCFDnCFDRFPCTRxLH: <RSCFDn_base> + H’00F9 + (H’04 × x), RSCFDnCFDRFPCTRxHL: <RSCFDn_base> + H’00FA + (H’04 × x), RSCFDnCFDRFPCTRxHH: <RSCFDn_base> + H’00FB + (H’04 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should be the value after reset.
7 to 0	RFPC[7:0]	All 0	W	Receive FIFO Pointer Control When these bits are set to H’FF, the read pointer moves to the next unread message in the receive FIFO buffer.

When the RFDMAEx value in the RSCFDnCFDCDTCT register is 1 (DMA transfer request enabled), do not write a value to this register.

RFPC[7:0] Bits

When the RFPC[7:0] bits are set to H’FF, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCFDnCFDRFSTSx register is decremented. Read the RSCFDnCFDRFIDx, RSCFDnCFDRFPTRx, RSCFDnCFDRFDSTSx, and RSCFDnCFDRFDFd_x registers to read messages in the receive FIFO buffer, and then write H’FF to the RFPC[7:0] bits. When writing H’FF to these bits, make sure that the RFE bit in the RSCFDnCFDRFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCFDnCFDRFSTSx register is 0 (the receive FIFO buffer contains unread messages).

28.4.7.4 RSCFDnCFDRFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access Size:	RSCFDnCFDRFIDx register can be read only in 32-bit units RSCFDnCFDRFIDxL, RSCFDnCFDRFIDxH registers can be read only in 16-bit units RSCFDnCFDRFIDxLL, RSCFDnCFDRFIDxLH, RSCFDnCFDRFIDxHL, RSCFDnCFDRFIDxHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDRFIDx: <RSCFDn_base> + H’3000 + (H’80 × x) RSCFDnCFDRFIDxL: <RSCFDn_base> + H’3000 + (H’80 × x), RSCFDnCFDRFIDxH: <RSCFDn_base> + H’3002 + (H’80 × x) RSCFDnCFDRFIDxLL: <RSCFDn_base> + H’3000 + (H’80 × x), RSCFDnCFDRFIDxLH: <RSCFDn_base> + H’3001 + (H’80 × x), RSCFDnCFDRFIDxHL: <RSCFDn_base> + H’3002 + (H’80 × x), RSCFDnCFDRFIDxHH: <RSCFDn_base> + H’3003 + (H’80 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFIDE	0	R	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	0	R	Receive FIFO Buffer RTR/RRS ● When the received message is a classical CAN frame 0: Data frame 1: Remote frame ● When the received message is a CANFD frame The RRS bit value of the received message can be read.
29	—	0	R	Reserved This bit is read as the value after reset.
28 to 0	RFID[28:0]	All 0	R	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

When the received message is a classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer. When the received message is a CANFD frame, this bit indicates the RRS bit value in the message.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

28.4.7.5 RSCFDnCFDRFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access Size: RSCFDnCFDRFPTRx register can be read only in 32-bit units

RSCFDnCFDRFPTRxL, RSCFDnCFDRFPTRxH registers can be read only in 16-bit units

RSCFDnCFDRFPTRxLL, RSCFDnCFDRFPTRxLH, RSCFDnCFDRFPTRxHL, RSCFDnCFDRFPTRxHH registers can be read only in 8- bit units

Address(es): RSCFDnCFDRFPTRx: <RSCFDn_base> + H’3004 + (H’80 × x)
RSCFDnCFDRFPTRxL: <RSCFDn_base> + H’3004 + (H’80 × x),
RSCFDnCFDRFPTRxH: <RSCFDn_base> + H’3006 + (H’80 × x)
RSCFDnCFDRFPTRxLL: <RSCFDn_base> + H’3004 + (H’80 × x),
RSCFDnCFDRFPTRxLH: <RSCFDn_base> + H’3005 + (H’80 × x),
RSCFDnCFDRFPTRxHL: <RSCFDn_base> + H’3006 + (H’80 × x),
RSCFDnCFDRFPTRxHH: <RSCFDn_base> + H’3007 + (H’80 × x)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RFDLC[3:0]	All 0	R	Receive FIFO Buffer DLC Data

b31	b30	b29	b28	Classical CAN Frame	CANFD Frame
0	0	0	0	0 data bytes	
0	0	0	1	1 data byte	
0	0	1	0	2 data bytes	
0	0	1	1	3 data bytes	
0	1	0	0	4 data bytes	
0	1	0	1	5 data bytes	
0	1	1	0	6 data bytes	
0	1	1	1	7 data bytes	
1	0	0	0	8 data bytes	
1	0	0	1	8 data bytes	12 data bytes
1	0	1	0		16 data bytes
1	0	1	1		20 data bytes
1	1	0	0		24 data bytes
1	1	0	1		32 data bytes
1	1	1	0		48 data bytes
1	1	1	1		64 data bytes

Bit	Bit Name	Initial Value	R/W	Description
27 to 16	RFPTR[11:0]	All 0	R	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS[15:0]	All 0	R	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

28.4.7.6 RSCFDnCFDRFFDSTSx — Receive FIFO CANFD Status Register (x = 0 to 7)

Access Size:	RSCFDnCFDRFFDSTSx register can be read only in 32-bit units RSCFDnCFDRFFDSTSxL, RSCFDnCFDRFFDSTSxH registers can be read only in 16-bit units RSCFDnCFDRFFDSTSxLL, RSCFDnCFDRFFDSTSxLH, RSCFDnCFDRFFDSTSxHL, RSCFDnCFDRFFDSTSxHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDRFFDSTSx: <RSCFDn_base> + H’3008 + (H’80 × x) RSCFDnCFDRFFDSTSxL: <RSCFDn_base> + H’3008 + (H’80 × x), RSCFDnCFDRFFDSTSxH: <RSCFDn_base> + H’300A + (H’80 × x) RSCFDnCFDRFFDSTSxLL: <RSCFDn_base> + H’3008 + (H’80 × x), RSCFDnCFDRFFDSTSxLH: <RSCFDn_base> + H’3009 + (H’80 × x), RSCFDnCFDRFFDSTSxHL: <RSCFDn_base> + H’300A + (H’80 × x), RSCFDnCFDRFFDSTSxHH: <RSCFDn_base> + H’300B + (H’80 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RFFDF	RFBRS	RFESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as the value after reset.
2	RFFDF	0	R	FDF 0: Classical CAN frame 1: CANFD frame
1	RFBRS	0	R	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	RFESI	0	R	ESI 0: Error active node 1: Error passive node

RFFDF Bit

This bit indicates the FD format (classical CAN frame or CANFD frame) of the message stored in the receive FIFO buffer.

RFBRS Bit

When the RFFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0.

RFESI Bit

When the RFFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0.

28.4.7.7 RSCFDnCFDRFDFd_x — Receive FIFO Buffer Access Data Field d Register (d = 0 to 15, x = 0 to 7)

Access Size:	RSCFDnCFDRFDFd_x register can be read only in 32-bit units RSCFDnCFDRFDFd_xL, RSCFDnCFDRFDFd_xH registers can be read only in 16-bit units RSCFDnCFDRFDFd_xLL, RSCFDnCFDRFDFd_xLH, RSCFDnCFDRFDFd_xHL, RSCFDnCFDRFDFd_xHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDRFDFd_x: <RSCFDn_base> + H’300C + (H’04 × d) + (H’80 × x) RSCFDnCFDRFDFd_xL: <RSCFDn_base> + H’300C + (H’04 × d) + (H’80 × x), RSCFDnCFDRFDFd_xH: <RSCFDn_base> + H’300E + (H’04 × d) + (H’80 × x) RSCFDnCFDRFDFd_xLL: <RSCFDn_base> + H’300C + (H’04 × d) + (H’80 × x), RSCFDnCFDRFDFd_xLH: <RSCFDn_base> + H’300D + (H’04 × d) + (H’80 × x), RSCFDnCFDRFDFd_xHL: <RSCFDn_base> + H’300E + (H’04 × d) + (H’80 × x), RSCFDnCFDRFDFd_xHH: <RSCFDn_base> + H’300F + (H’04 × d) + (H’80 × x)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB4 × d + 3 [7:0]								RFDB4 × d + 2 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB4 × d + 1 [7:0]								RFDB4 × d + 0 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RFDB4 × d + 3 [7:0]	All 0	R	Receive Buffer Data Byte 4 × d + 3 Receive Buffer Data Byte 4 × d + 2 Receive Buffer Data Byte 4 × d + 1 Receive Buffer Data Byte 4 × d + 0 Data for a message stored in the receive buffer can be read.
23 to 16	RFDB4 × d + 2 [7:0]	All 0	R	
15 to 8	RFDB4 × d + 1 [7:0]	All 0	R	
7 to 0	RFDB4 × d + 0 [7:0]	All 0	R	

When the RFDLC[3:0] value in the RSCFDnCFDRFPTRx register is smaller than the payload storage size of the receive FIFO buffer, data bytes for which no data is set are read as H’00.

Specify the payload storage size of the receive FIFO buffer by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register.

Do not read or write the RSCFDnCFDRFDFd_x register corresponding to an area larger than the specified size.

28.4.8 Transmit/Receive FIFO Buffer-related Registers

28.4.8.1 RSCFDnCFDCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)

Access Size:	RSCFDnCFDCFCCk register can be read/written in 32-bit units RSCFDnCFDCFCCkL, RSCFDnCFDCFCCkH registers can be read/written in 16-bit units RSCFDnCFDCFCCkLL, RSCFDnCFDCFCCkLH, RSCFDnCFDCFCCkHL, RSCFDnCFDCFCCkHH registers can be read/written in 8- bit units
Address(es):	RSCFDnCFDCFCCk: <RSCFDn_base> + H’0118 + (H’04 × k) RSCFDnCFDCFCCkL: <RSCFDn_base> + H’0118 + (H’04 × k), RSCFDnCFDCFCCkH: <RSCFDn_base> + H’011A + (H’04 × k) RSCFDnCFDCFCCkLL: <RSCFDn_base> + H’0118 + (H’04 × k), RSCFDnCFDCFCCkLH: <RSCFDn_base> + H’0119 + (H’04 × k), RSCFDnCFDCFCCkHL: <RSCFDn_base> + H’011A + (H’04 × k), RSCFDnCFDCFCCkHH: <RSCFDn_base> + H’011B + (H’04 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	CFPLS[2:0]			—	CFTXIE	CFRXIE	CFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFITT[7:0]	All 0	R/W	Set a message transmission interval. Set Value: H’00 to H’FF
23 to 20	CFTML[3:0]	All 0	R/W	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	0	R/W	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk by (ITRCP[15:0] bits) 1: Clock dividing pclk by (ITRCP[15:0] bits × 10)
18	CFITSS	0	R/W	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the nominal bit time clock for the channel to which the FIFO is linked.
17 to 16	CFM[1:0]	All 0	R/W	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	CFIGCV[2:0]	All 0	R/W	Transmit/Receive FIFO Receive Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	CFIM	0	R/W	Transmit/Receive FIFO Interrupt Source Select 0: ● Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. ● Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. 1: ● Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. ● Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.
11	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
10 to 8	CFDC[2:0]	All 0	R/W	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
6 to 4	CFPLS[2:0]	All 0	R/W	Transmit/Receive FIFO Buffer Payload Storage Size Select b6 b5 b4 0 0 0: 8 bytes 0 0 1: 12 bytes 0 1 0: 16 bytes 0 1 1: 20 bytes 1 0 0: 24 bytes 1 0 1: 32 bytes 1 1 0: 48 bytes 1 1 1: 64 bytes
3	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.

Bit	Bit Name	Initial Value	R/W	Description
2	CFTXIE	0	R/W	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
1	CFRXIE	0	R/W	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
0	CFE	0	R/W	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01b (transmit mode) or 10b (gateway mode).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01b (transmit mode) or 10b (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as m = k/3 (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be ((16 × m) + CFTML[3:0]).

See **Table 28.11** and **Table 28.12**, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p. Setting the CFDC[2:0] bits to 001b or more enables the setting of the CFTML[3:0] bits.

Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFITR Bit

This bit is enabled when the CFITSS bit is 0.

When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register.

When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register × 10).

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.

When this bit is 1, the nominal bit time clock of the channel to which the FIFO is linked is the count source of the interval timer. Use this count source only for the channel that does not handle the CANFD frames.

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFM[1:0] Bits

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to 00b (receive mode) or 10b (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).

When the CFDC[2:0] bits are set to 001b (4 messages), set the CFIGCV[2:0] bits to 001b, 011b, 101b, or 111b.

Modify these bits only in global reset mode.

CFIM Bit

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFDC[2:0] Bits

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000b, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFPLS[2:0] Bits

These bits are used to select the maximum payload size that can be stored in the transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFRXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.

Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.

When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/ receive FIFO buffer becomes empty immediately.

This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

After all other bits in the RSCFDnCFDCFCCk register have been set, set this bit to 1 by using another instruction.

28.4.8.2 RSCFDnCFDCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)

Access Size:	RSCFDnCFDCFSTSk register can be read/written in 32-bit units RSCFDnCFDCFSTSkL, RSCFDnCFDCFSTSkH registers can be read/written in 16-bit units RSCFDnCFDCFSTSkLL, RSCFDnCFDCFSTSkLH, RSCFDnCFDCFSTSkHL, RSCFDnCFDCFSTSkHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDCFSTSk: <RSCFDn_base> + H’0178 + (H’04 × k) RSCFDnCFDCFSTSkL: <RSCFDn_base> + H’0178 + (H’04 × k), RSCFDnCFDCFSTSkH: <RSCFDn_base> + H’017A + (H’04 × k) RSCFDnCFDCFSTSkLL: <RSCFDn_base> + H’0178 + (H’04 × k), RSCFDnCFDCFSTSkLH: <RSCFDn_base> + H’0179 + (H’04 × k), RSCFDnCFDCFSTSkHL: <RSCFDn_base> + H’017A + (H’04 × k), RSCFDnCFDCFSTSkHH: <RSCFDn_base> + H’017B + (H’04 × k)
Initial Value:	H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
15 to 8	CFMC[7:0]	All 0	R	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset. When writing to these bits, write the value after reset.
4	CFTXIF	0	R/W*1	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.
3	CFRXIF	0	R/W*1	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	0	R/W*1	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	0	R	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	0	R	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCFDnCFDCFCCk register.

- When CFM[1:0] value is 01b (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00b (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10b (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00b: In global reset mode
- When CFM[1:0] value is 01b or 10b: In channel reset mode
- When the CFE bit in the RSCFDnCFDCFCCk register is cleared to 0.

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01b or 10b, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00b or 10b, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

Write 0 to this flag in global operating mode or global test mode

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCFDnCFDCFCCk register.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCFDnCFDCFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00b: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00b or 10b: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01b: A value of H’FF has been written to the RSCFDnCFDCFPCTRk register after data was written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers.

NOTE

To clear CFTXIF, CFRXIF, or CFMLT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

28.4.8.3 RSCFDnCFDCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)

Access Size:	RSCFDnCFDCFPCTRk register can only be written in 32-bit units RSCFDnCFDCFPCTRkL, RSCFDnCFDCFPCTRkH registers can only be written in 16-bit units RSCFDnCFDCFPCTRkLL, RSCFDnCFDCFPCTRkLH, RSCFDnCFDCFPCTRkHL, RSCFDnCFDCFPCTRkHH registers can only be written in 8-bit units
Address(es):	RSCFDnCFDCFPCTRk: <RSCFDn_base> + H'01D8 + (H'04 × k) RSCFDnCFDCFPCTRkL: <RSCFDn_base> + H'01D8 + (H'04 × k), RSCFDnCFDCFPCTRkH: <RSCFDn_base> + H'01DA + (H'04 × k) RSCFDnCFDCFPCTRkLL: <RSCFDn_base> + H'01D8 + (H'04 × k), RSCFDnCFDCFPCTRkLH: <RSCFDn_base> + H'01D9 + (H'04 × k), RSCFDnCFDCFPCTRkHL: <RSCFDn_base> + H'01DA + (H'04 × k), RSCFDnCFDCFPCTRkHH: <RSCFDn_base> + H'01DB + (H'04 × k)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should be the value after reset.
7 to 0	CFPC[7:0]	All 0	W	Transmit/Receive FIFO Pointer Control • Receive mode: Writing H'FF to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing H'FF to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited

When the corresponding transmit/receive FIFO buffer is the first transmit/receive FIFO buffer (k = 3 × m) allocated to channel m and when the CFDMAEm bit in the RSCFDnCFDCDTCT register is 1 (DMA transfer request enabled), do not write a value to this register.

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 00b):
  Writing H'FF to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCFDnCFDCFSTSk register is decremented. Read the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers to read messages from the transmit/receive FIFO buffer, and then write H'FF to the CFPC[7:0] bits.
  When writing H'FF to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 01b):
  Writing H'FF to the CFPC[7:0] bits stores the data written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers before writing H'FF to the CFPC[7:0] bits.
  When writing H'FF to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCk register is set to 1 and the CFFLL flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 10b):
  Setting prohibited

28.4.8.4 RSCFDnCFDCFIDk — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)

Access Size:	RSCFDnCFDCFIDk register can be read/written in 32-bit units RSCFDnCFDCFIDkL, RSCFDnCFDCFIDkH registers can be read/written in 16-bit units RSCFDnCFDCFIDkLL, RSCFDnCFDCFIDkLH, RSCFDnCFDCFIDkHL, RSCFDnCFDCFIDkHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDCFIDk: <RSCFDn_base> + H’3400 + (H’80 × k) RSCFDnCFDCFIDkL: <RSCFDn_base> + H’3400 + (H’80 × k), RSCFDnCFDCFIDkH: <RSCFDn_base> + H’3402 + (H’80 × k) RSCFDnCFDCFIDkLL: <RSCFDn_base> + H’3400 + (H’80 × k), RSCFDnCFDCFIDkLH: <RSCFDn_base> + H’3401 + (H’80 × k), RSCFDnCFDCFIDkHL: <RSCFDn_base> + H’3402 + (H’80 × k), RSCFDnCFDCFIDkHH: <RSCFDn_base> + H’3403 + (H’80 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CFIDE	0	R/W	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	CFRTR	0	R/W	Transmit/Receive FIFO Buffer RTR/RRS ● When the CFM[1:0] value is 01b (transmit mode) – When the transmit message is a classical CAN frame 0: Data frame 1: Remote frame – When the transmit message is a CANFD frame Write 0 to this bit. ● When the CFM[1:0] value is 00b (receive mode) – When the received message is a classical CAN frame 0: Data frame 1: Remote frame – When the received message is a CANFD frame The RRS bit value of the received message can be read.
29	THLEN	0	R/W	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01b (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.

Bit	Bit Name	Initial Value	R/W	Description
28 to 0	CFID[28:0]	All 0	R/W	Transmit/Receive FIFO Buffer ID Data ● When CFM[1:0] value is 01b (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. ● When CFM[1:0] value is 00b (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0.

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). This RSCFDnCFDCFIDk register should not be read or written when the CFM[1:0] value is 10b (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b. When the CFM[1:0] value is 01b, this bit is used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

If the received message is a classical CAN frame, this bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b. If the received message is a CANFD frame, this bit indicates the RRS bit value of the received message.

When the CFM[1:0] value is 01b, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

When the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 1 (CANFD frame), set this bit to 0.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp) of transmit messages is stored in the transmit history buffer after transmission is completed.

This bit is enabled when the CFM[1:0] value is 01b (transmit mode).

CFID[28:0] Bits

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b.

When the CFM[1:0] value is 01b, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

28.4.8.5 RSCFDnCFDCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5)

Access Size:	RSCFDnCFDCFPTRk register can be read/written in 32-bit units RSCFDnCFDCFPTRkL, RSCFDnCFDCFPTRkH registers can be read/written in 16-bit units RSCFDnCFDCFPTRkLL, RSCFDnCFDCFPTRkLH, RSCFDnCFDCFPTRkHL, RSCFDnCFDCFPTRkHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDCFPTRk: <RSCFDn_base> + H’3404 + (H’80 × k) RSCFDnCFDCFPTRkL: <RSCFDn_base> + H’3404 + (H’80 × k), RSCFDnCFDCFPTRkH: <RSCFDn_base> + H’3406 + (H’80 × k) RSCFDnCFDCFPTRkLL: <RSCFDn_base> + H’3404 + (H’80 × k), RSCFDnCFDCFPTRkLH: <RSCFDn_base> + H’3405 + (H’80 × k), RSCFDnCFDCFPTRkHL: <RSCFDn_base> + H’3406 + (H’80 × k), RSCFDnCFDCFPTRkHH: <RSCFDn_base> + H’3407 + (H’80 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>30 to 28</td><td>CFDLC[3:0]</td><td>All 0</td><td>R/W</td><td>Transmit/Receive FIFO Buffer DLC Data
<table>
<tr><th>b31</th><th>b30</th><th>b29</th><th>b28</th><th>Classical CAN Frame</th><th>CANFD Frame</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0 data bytes</td><td></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1 data byte</td><td></td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>2 data bytes</td><td></td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>3 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>4 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>5 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>6 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>7 data bytes</td><td></td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>8 data bytes</td><td></td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>8 data bytes</td><td>12 data bytes</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td></td><td>16 data bytes</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td></td><td>20 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td></td><td>24 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td></td><td>32 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td></td><td>48 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td></td><td>64 data bytes</td></tr>
</table></td></tr>
<tr><td>27 to 16</td><td>CFPTR[11:0]</td><td>All 0</td><td>R/W</td><td>Transmit/Receive FIFO Buffer Label Data
● When CFM[1:0] value is 01b (transmit mode):
Set the label information to be stored in the transmit history buffer.
Only bits CFPTR[7:0] are valid.
● When CFM[1:0] value is 00b (receive mode):
The label information of the received message can be read.</td></tr>
</table>

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CFTS[15:0]	All 0	R/W	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00b (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). This register should not be read or written when the CFM[1:0] value is 10b (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b.

When the CFM[1:0] value is 01b, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer.

When the CFDLC[3:0] bits are set to 1001b or more while the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 0 (CAN frame), 8-byte data is transmitted actually. When the CFFDF bit is 1 (CANFD frame), the settable value range varies depending on the settings of the TMME bit in the RSCFDnCFDCmFDCFG register and the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

- When TMME bit = 0 (transmit buffer merge mode disabled):
  A value of 0000b to 1111b is settable. If the specified data length exceeds the payload storage size specified by the CFPLS[2:0] bits, excessive payloads are padded by H’CC.
- When TMME bit = 1 (transmit buffer merge mode enabled):
  Set the data length within the payload storage size specified by the CFPLS[2:0] bits.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b. When the CFM[1:0] value is 01b, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer. These bits are valid when the CFM[1:0] value is 00b.

28.4.8.6 RSCFDnCFDCFFDCSTSk — Transmit/Receive FIFO CANFD Configuration/ Status Register (k = 0 to 5)

Access Size:	RSCFDnCFDCFFDCSTSk register can be read/written in 32-bit units RSCFDnCFDCFFDCSTSkL, RSCFDnCFDCFFDCSTSkH registers can be read/written in 16-bit units RSCFDnCFDCFFDCSTSkLL, RSCFDnCFDCFFDCSTSkLH, RSCFDnCFDCFFDCSTSkHL, RSCFDnCFDCFFDCSTSkHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDCFFDCSTSk: <RSCFDn_base> + H’3408 + (H’80 × k) RSCFDnCFDCFFDCSTSkL: <RSCFDn_base> + H’3408 + (H’80 × k), RSCFDnCFDCFFDCSTSkH: <RSCFDn_base> + H’340A + (H’80 × k) RSCFDnCFDCFFDCSTSkLL: <RSCFDn_base> + H’3408 + (H’80 × k), RSCFDnCFDCFFDCSTSkLH: <RSCFDn_base> + H’3409 + (H’80 × k), RSCFDnCFDCFFDCSTSkHL: <RSCFDn_base> + H’340A + (H’80 × k), RSCFDnCFDCFFDCSTSkHH: <RSCFDn_base> + H’340B + (H’80 × k)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CFFDF	CFBRS	CFESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as the value after reset.
2	CFFDF	0	R/W	FDF 0: Classical CAN frame 1: CANFD frame
1	CFBRS	0	R/W	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	CFESI	0	R/W	ESI 0: Error active node 1: Error passive node

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). Do not read or write this register when the CFM[1:0] value is 10b (gateway mode).

CFFDF Bit

When the CFM[1:0] value is 00b, this bit indicates the FD format (classical CAN frame or CANFD frame) of the message stored in the transmit/receive FIFO buffer. When the CFM[1:0] value is 01b, this bit is used to set the FD format of the message to be transmitted from the transmit/receive FIFO buffer.

CFBRS Bit

When the CFM[1:0] value is 00b, if the CFFDF bit is 1, this bit indicates the BRS bit value of the message stored in the transmit/receive FIFO buffer. If the CFFDF bit is 0, this bit is always read as 0.

When the CFM[1:0] value is 01b, if the CFFDF bit is 1, this bit is used to set the BRS bit value of the message to be transmitted from the transmit/receive FIFO buffer. If the CFFDF bit is 0, write 0 to this bit.

CFESI Bit

When the CFM[1:0] value is 00b, if the CFFDF bit is 1, this bit indicates the ESI bit value of the message stored in the transmit/receive FIFO buffer. If the CFFDF bit is 0, this bit is always read as 0.

When the CFM[1:0] value is 01b, if the CFFDF bit is 1, this bit is used to set the ESI bit value of the message to be transmitted from the transmit/receive FIFO buffer. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in the error active state. When the channel is in the error passive state, the ESI bit value that shows an error passive node is transmitted regardless of this bit value. When the CFFDF bit is 0, write 0 to this bit.

28.4.8.7 RSCFDnCFDCFDFd_k — Transmit/receive FIFO Buffer Access Data Field d Register (d = 0 to 15, k = 0 to 5)

Access Size: RSCFDnCFDCFDFd_k register can be read/written in 32-bit units
RSCFDnCFDCFDFd_kL, RSCFDnCFDCFDFd_kH registers can be read/written in 16-bit units
RSCFDnCFDCFDFd_kLL, RSCFDnCFDCFDFd_kLH, RSCFDnCFDCFDFd_kHL, RSCFDnCFDCFDFd_kHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDCFDFd_k: <RSCFDn_base> + H’340C + (H’04 × d) + (H’80 × k)
RSCFDnCFDCFDFd_kL: <RSCFDn_base> + H’340C + (H’04 × d) + (H’80 × k),
RSCFDnCFDCFDFd_kH: <RSCFDn_base> + H’340E + (H’04 × d) + (H’80 × k)
RSCFDnCFDCFDFd_kLL: <RSCFDn_base> + H’340C + (H’04 × d) + (H’80 × k),
RSCFDnCFDCFDFd_kLH: <RSCFDn_base> + H’340D + (H’04 × d) + (H’80 × k),
RSCFDnCFDCFDFd_kHL: <RSCFDn_base> + H’340E + (H’04 × d) + (H’80 × k),
RSCFDnCFDCFDFd_kHH: <RSCFDn_base> + H’340F + (H’04 × d) + (H’80 × k)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB4 × d + 3 [7:0]								CFDB4 × d + 2 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB4 × d + 1 [7:0]								CFDB4 × d + 1 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFDB4 × d + 3 [7:0]	All 0	R/W	Transmit/Receive FIFO Buffer Data Byte 4 × d + 3 Transmit/Receive FIFO Buffer Data Byte 4 × d + 2 Transmit/Receive FIFO Buffer Data Byte 4 × d + 1 Transmit/Receive FIFO Buffer Data Byte 4 × d + 0 ● When CFM[1:0] value is 01b (transmit mode): Set the transmit/receive FIFO buffer data. ● When CFM[1:0] value is 00b (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.
23 to 16	CFDB4 × d + 2 [7:0]	All 0	R/W	
15 to 8	CFDB4 × d + 1 [7:0]	All 0	R/W	
7 to 0	CFDB4 × d + 0 [7:0]	All 0	R/W	

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01b (transmit mode). This register is readable only when the CFM[1:0] value is 00b (receive mode). When the CFDLC[3:0] value in the RSCFDnCFDCFPTRk register is smaller than the payload storage size of the transmit/receive FIFO buffer, data bytes for which no data is set are read as H’00.

Specify the payload storage size of the transmit/receive FIFO buffer by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register. Do not read or write the RSCFDnCFDCFDFd_k register corresponding to an area larger than the specified size.

This register should not be read or written when the CFM[1:0] value is 10b (gateway mode).

28.4.9 Details of FIFO Status-related Registers

28.4.9.1 RSCFDnCFDFESTS — FIFO Empty Status Register

Access Size:	RSCFDnCFDFESTS register can be read only in 32-bit units RSCFDnCFDFESTSL, RSCFDnCFDFESTSH registers can be read only in 16-bit units RSCFDnCFDFESTSLL, RSCFDnCFDFESTSLH, RSCFDnCFDFESTSHL, RSCFDnCFDFESTSHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDFESTS: <RSCFDn_base> + H’0238 RSCFDnCFDFESTSL: <RSCFDn_base> + H’0238, RSCFDnCFDFESTSH: <RSCFDn_base> + H’023A RSCFDnCFDFESTSLL: <RSCFDn_base> + H’0238, RSCFDnCFDFESTSLH: <RSCFDn_base> + H’0239, RSCFDnCFDFESTSHL: <RSCFDn_base> + H’023A, RSCFDnCFDFESTSHH: <RSCFDn_base> + H’023B
Initial Value:	H’03FF FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5EMP	CF4EMP	CF3EMP	CF2EMP	CF1EMP	CF0EMP	RF7EMP	RF6EMP	RF5EMP	RF4EMP	RF3EMP	RF2EMP	RF1EMP	RF0EMP
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	H’0_0FFF	R	Reserved These bits are read as the value after reset.
13	CF5EMP	1	R	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 5)
12	CF4EMP	1	R	
11	CF3EMP	1	R	
10	CF2EMP	1	R	
9	CF1EMP	1	R	
8	CF0EMP	1	R	
7	RF7EMP	1	R	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message (buffer empty). (x = 0 to 7)
6	RF6EMP	1	R	
5	RF5EMP	1	R	
4	RF4EMP	1	R	
3	RF3EMP	1	R	
2	RF2EMP	1	R	
1	RF1EMP	1	R	
0	RF0EMP	1	R	

This RSCFDnCFDFESTS register is set to 03FF H’FFFF in global reset mode.

CFkEMP Flag (k = 0 to 5)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCFDnCFDCFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

28.4.9.2 RSCFDnCFDFFSTS — FIFO Full Status Register

Access Size: RSCFDnCFDFFSTS register can be read only in 32-bit units
RSCFDnCFDFFSTSL, RSCFDnCFDFFSTSH registers can be read only in 16-bit units
RSCFDnCFDFFSTSLL, RSCFDnCFDFFSTSLH, RSCFDnCFDFFSTSHL, RSCFDnCFDFFSTSHH registers can be read only in 8-bit units

Address(es): RSCFDnCFDFFSTS: <RSCFDn_base> + H'023C
RSCFDnCFDFFSTSL: <RSCFDn_base> + H'023C, RSCFDnCFDFFSTSH: <RSCFDn_base> + H'023E
RSCFDnCFDFFSTSLL: <RSCFDn_base> + H'023C, RSCFDnCFDFFSTSLH: <RSCFDn_base> + H'023D,
RSCFDnCFDFFSTSHL: <RSCFDn_base> + H'023E, RSCFDnCFDFFSTSHH: <RSCFDn_base> + H'023F

Initial Value: H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5FLL	CF4FLL	CF3FLL	CF2FLL	CF1FLL	CF0FLL	RF7FLL	RF6FLL	RF5FLL	RF4FLL	RF3FLL	RF2FLL	RF1FLL	RF0FLL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset.
13	CF5FLL	0	R	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 0 to 5)
12	CF4FLL	0	R	
11	CF3FLL	0	R	
10	CF2FLL	0	R	
9	CF1FLL	0	R	
8	CF0FLL	0	R	
7	RF7FLL	0	R	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7)
6	RF6FLL	0	R	
5	RF5FLL	0	R	
4	RF4FLL	0	R	
3	RF3FLL	0	R	
2	RF2FLL	0	R	
1	RF1FLL	0	R	
0	RF0FLL	0	R	

The RSCFDnCFDFFSTS register is cleared to H'0000 0000 in global reset mode.

CFkFLL Flag (k = 0 to 5)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCFDnCFDCFSTSk register is set to 1 (the transmit/receive FIFO buffer is full).

When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

28.4.9.3 RSCFDnCFDFMSTS — FIFO Message Lost Status Register

Access Size:	RSCFDnCFDFMSTS register can be read only in 32-bit units RSCFDnCFDFMSTSL, RSCFDnCFDFMSTSH registers can be read only in 16-bit units RSCFDnCFDFMSTSLL, RSCFDnCFDFMSTSLH, RSCFDnCFDFMSTSHL, RSCFDnCFDFMSTSHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDFMSTS: <RSCFDn_base> + H’0240 RSCFDnCFDFMSTSL: <RSCFDn_base> + H’0240, RSCFDnCFDFMSTSH: <RSCFDn_base> + H’0242 RSCFDnCFDFMSTSLL: <RSCFDn_base> + H’0240, RSCFDnCFDFMSTSLH: <RSCFDn_base> + H’0241, RSCFDnCFDFMSTSHL: <RSCFDn_base> + H’0242, RSCFDnCFDFMSTSHH: <RSCFDn_base> + H’0243
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CF5MLT	CF4MLT	CF3MLT	CF2MLT	CF1MLT	CF0MLT	RF7MLT	RF6MLT	RF5MLT	RF4MLT	RF3MLT	RF2MLT	RF1MLT	RF0MLT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset.
13	CF5MLT	0	R	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 5)
12	CF4MLT	0	R	
11	CF3MLT	0	R	
10	CF2MLT	0	R	
9	CF1MLT	0	R	
8	CF0MLT	0	R	
7	RF7MLT	0	R	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	0	R	
5	RF5MLT	0	R	
4	RF4MLT	0	R	
3	RF3MLT	0	R	
2	RF2MLT	0	R	
1	RF1MLT	0	R	
0	RF0MLT	0	R	

The RSCFDnCFDFMSTS register is cleared to H’0000 0000 in global reset mode.

CFkMLT Flag (k = 0 to 5)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCFDnCFDRFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

28.4.9.4 RSCFDnCFDRFISTS — Receive FIFO Buffer Interrupt Flag Status Register

Access Size:	RSCFDnCFDRFISTS register can be read only in 32-bit units RSCFDnCFDRFISTSL, RSCFDnCFDRFISTSH registers can be read only in 16-bit units RSCFDnCFDRFISTSLL, RSCFDnCFDRFISTSLH, RSCFDnCFDRFISTSHL, RSCFDnCFDRFISTSHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDRFISTS: <RSCFDn_base> + H'0244 RSCFDnCFDRFISTSL: <RSCFDn_base> + H'0244, RSCFDnCFDRFISTSH: <RSCFDn_base> + H'0246 RSCFDnCFDRFISTSLL: <RSCFDn_base> + H'0244, RSCFDnCFDRFISTSLH: <RSCFDn_base> + H'0245, RSCFDnCFDRFISTSHL: <RSCFDn_base> + H'0246, RSCFDnCFDRFISTSHH: <RSCFDn_base> + H'0247
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are read as the value after reset.
7	RF7IF	0	R	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	0	R	
5	RF5IF	0	R	
4	RF4IF	0	R	
3	RF3IF	0	R	
2	RF2IF	0	R	
1	RF1IF	0	R	
0	RF0IF	0	R	

The RSCFDnCFDRFISTS register is cleared to H'0000 0000 in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCFDnCFDRFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

28.4.9.5 RSCFDnCFDCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register

Access Size:	RSCFDnCFDCFRISTS register can be read only in 32-bit units RSCFDnCFDCFRISTSL, RSCFDnCFDCFRISTSH registers can be read only in 16-bit units RSCFDnCFDCFRISTSLL, RSCFDnCFDCFRISTSLH, RSCFDnCFDCFRISTSHL, RSCFDnCFDCFRISTSHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDCFRISTS: <RSCFDn_base> + H’0248 RSCFDnCFDCFRISTSL: <RSCFDn_base> + H’0248, RSCFDnCFDCFRISTSH: <RSCFDn_base> + H’024A RSCFDnCFDCFRISTSLL: <RSCFDn_base> + H’0248, RSCFDnCFDCFRISTSLH: <RSCFDn_base> + H’0249, RSCFDnCFDCFRISTSHL: <RSCFDn_base> + H’024A, RSCFDnCFDCFRISTSHH: <RSCFDn_base> + H’024B
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5RXIF	CF4RXIF	CF3RXIF	CF2RXIF	CF1RXIF	CF0RXIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are read as the value after reset.
5	CF5RXIF	0	R	Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k receive interrupt request is present. 1: A transmit/receive FIFO buffer k receive interrupt request is present. (k = 0 to 5)
4	CF4RXIF	0	R	
3	CF3RXIF	0	R	
2	CF2RXIF	0	R	
1	CF1RXIF	0	R	
0	CF0RXIF	0	R	

The RSCFDnCFDCFRISTS register is cleared to H’0000 0000 in global reset mode.

CFkRXIF Flag (k = 0 to 5)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

28.4.9.6 RSCFDnCFDCFTISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register

Access Size:	RSCFDnCFDCFTISTS register can be read only in 32-bit units RSCFDnCFDCFTISTSL, RSCFDnCFDCFTISTSH registers can be read only in 16-bit units RSCFDnCFDCFTISTSLL, RSCFDnCFDCFTISTSLH, RSCFDnCFDCFTISTSHL, RSCFDnCFDCFTISTSHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDCFTISTS: <RSCFDn_base> + H’024C RSCFDnCFDCFTISTSL: <RSCFDn_base> + H’024C, RSCFDnCFDCFTISTSH: <RSCFDn_base> + H’024E RSCFDnCFDCFTISTSLL: <RSCFDn_base> + H’024C, RSCFDnCFDCFTISTSLH: <RSCFDn_base> + H’024D, RSCFDnCFDCFTISTSHL: <RSCFDn_base> + H’024E, RSCFDnCFDCFTISTSHH: <RSCFDn_base> + H’024F
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CF5TXIF	CF4TXIF	CF3TXIF	CF2TXIF	CF1TXIF	CF0TXIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are read as the value after reset.
5	CF5TXIF	0	R	Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. (k = 0 to 5)
4	CF4TXIF	0	R	
3	CF3TXIF	0	R	
2	CF2TXIF	0	R	
1	CF1TXIF	0	R	
0	CF0TXIF	0	R	

The RSCFDnCFDCFTISTS register is cleared to H’0000 0000 in global reset mode.

CFkTXIF Flag (k = 0 to 5)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

28.4.10 Details of FIFO DMA-related Registers

28.4.10.1 RSCFDnCFDCDTCT — DMA Enable Register

Access Size: RSCFDnCFDCDTCT register can be read/written in 32-bit units
RSCFDnCFDCDTCTL, RSCFDnCFDCDTCTH registers can be read/written in 16-bit units
RSCFDnCFDCDTCTLL, RSCFDnCFDCDTCTLH, RSCFDnCFDCDTCTHL, RSCFDnCFDCDTCTHH registers can be read/written in 8- bit units

Address(es): RSCFDnCFDCDTCT: <RSCFDn_base> + H’0490
RSCFDnCFDCDTCTL: <RSCFDn_base> + H’0490
RSCFDnCFDCDTCTH: <RSCFDn_base> + H’0492
RSCFDnCFDCDTCTLL: <RSCFDn_base> + H’0490, RSCFDnCFDCDTCTLH: <RSCFDn_base> + H’0491,
RSCFDnCFDCDTCTHL: <RSCFDn_base> + H’0492, RSCFDnCFDCDTCTHH: <RSCFDn_base> + H’0493

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CFDMAE1	CFDMAE0	RFDMAE7	RFDMAE6	RFDMAE5	RFDMAE4	RFDMAE3	RFDMAE2	RFDMAE1	RFDMAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
9	CFDMAE1	0	R/W	Transmit/Receive FIFO Buffer 3 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 3 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 3 is enabled.
8	CFDMAE0	0	R/W	Transmit/Receive FIFO Buffer 0 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 0 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 0 is enabled.
7	RFDMAE7	0	R/W	Receive FIFO Buffer x DMA Enable 0: A DMA transfer request of receive FIFO buffer x is disabled. 1: A DMA transfer request of receive FIFO buffer x is enabled. (x = 0 to 7)
6	RFDMAE6	0	R/W	
5	RFDMAE5	0	R/W	
4	RFDMAE4	0	R/W	
3	RFDMAE3	0	R/W	
2	RFDMAE2	0	R/W	
1	RFDMAE1	0	R/W	
0	RFDMAE0	0	R/W	

Modify the RSCFDnCFDCDTCT register in global operating mode or global test mode.

CFDMAEm Bit

This bit is used to enable DMA transfer for transmit/receive FIFO buffer 3 × m (the first transmit/receive FIFO buffer allocated to channel m). DMA transfer is enabled only for transmit/receive FIFO buffers for which the CFM[1:0] bits in the RSCFDnCFDCFCCk register is set to 00b (receive mode). Set this bit to 0 when the CFM[1:0] value is 01b (transmit mode) or 10b (gateway mode).

RFDMAEx Bit

This bit is used to enable DMA transfer for receive FIFO buffer x.

28.4.10.2 RSCFDnCFDCDTSTS — DMA Status Register

Access Size:	RSCFDnCFDCDTSTS register can be read only in 32-bit units RSCFDnCFDCDTSTSL, RSCFDnCFDCDTSTSH registers can be read only in 16-bit units RSCFDnCFDCDTSTSLL, RSCFDnCFDCDTSTSLH, RSCFDnCFDCDTSTSHL, RSCFDnCFDCDTSTSHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDCDTSTS: <RSCFDn_base> + H'0494 RSCFDnCFDCDTSTSL: <RSCFDn_base> + H'0494 RSCFDnCFDCDTSTSH: <RSCFDn_base> + H'0496 RSCFDnCFDCDTSTSLL: <RSCFDn_base> + H'0494, RSCFDnCFDCDTSTSLH: <RSCFDn_base> + H'0495, RSCFDnCFDCDTSTSHL: <RSCFDn_base> + H'0496, RSCFDnCFDCDTSTSHH: <RSCFDn_base> + H'0497
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CFDMASTS1	CFDMASTS0	RFDMASTS7	RFDMASTS6	RFDMASTS5	RFDMASTS4	RFDMASTS3	RFDMASTS2	RFDMASTS1	RFDMAASTS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are read as the value after reset.
9	CFDMASTS1	0	R	Transmit/Receive FIFO Buffer 3 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 3 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 3 is in progress.
8	CFDMASTS0	0	R	Transmit/Receive FIFO Buffer 0 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 0 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 0 is in progress.
7	RFDMASTS7	0	R	Receive FIFO Buffer x DMA Status 0: DMA transfer of receive FIFO buffer x is not in progress. 1: DMA transfer of receive FIFO buffer x is in progress. (x = 0 to 7)
6	RFDMASTS6	0	R	
5	RFDMASTS5	0	R	
4	RFDMASTS4	0	R	
3	RFDMASTS3	0	R	
2	RFDMASTS2	0	R	
1	RFDMASTS1	0	R	
0	RFDMASTS0	0	R	

CFDMASTSm Bit

When DMA transfer is enabled (CFDMAEm bit in the RSCFDnCFDCDTCT register is 1) for the transmit/receive FIFO buffer 3 × m (the first transmit/receive FIFO buffer allocated to channel m) while the transmit/receive FIFO buffer contains one of more messages, the CFDMASTSm bit is set to 1 indicating that DMA transfer is in progress.

When all messages in the transmit/receive FIFO buffer have been transferred or DMA transfer is disabled (CFDMAEm bit is 0), the CFDMASTSm bit is cleared to 0 indicating that DMA transfer has been completed. If the CFDMAEm bit is set to 0 during DMA transfer, the CFDMASTSm bit is cleared to 0 after the ongoing DMA transfer has been completed (when the message that is being transferred has been transferred to the last byte in the payload storage area).

These bits are cleared to 0 in global reset mode.

RFDMASTSx Bit

When DMA transfer is enabled (corresponding RFDMAEx bit in the RSCFDnCFDCDTCT register is 1) for the receive FIFO buffer x and the receive FIFO buffer contains one of more messages, the RFDMAEx bit is set to 1 indicating that DMA transfer is in progress.

When all messages in the receive FIFO buffer x have been transferred or DMA transfer is disabled (RFDMAEx bit = 0), the RFDMASTSx bit is cleared to 0 indicating that DMA transfer has been completed. If the RFDMAEx bit is set to 0 during DMA transfer, the RFDMASTSx bit is cleared to 0 after the ongoing DMA transfer has been completed (when the message that is being transferred has been transferred to the last byte in the payload storage area)

These bits are cleared to 0 in global reset mode.

28.4.11 Details of Transmit Buffer-related Registers

28.4.11.1 RSCFDnCFDTMCp — Transmit Buffer Control Register (p = 0 to 31)

Access Size: RSCFDnCFDTMCp registers can be read/written in 8-bit units

Address(es): RSCFDnCFDTMCp: <RSCFDn_base> + H'0250 + (H'01 × p)

Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	TMOM	0	R/W	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	0	R/W*1	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	0	R/W*1	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

When the RSCFDnCFDTMCp register meets any of the following conditions, set it to H'00.

- The RSCFDnCFDTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCFDnCFDCFCCk register (p = m × 16 + the value of CFTML[3:0] bits).
- The RSCFDnCFDTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCFDnCFDTXQCCm (m = 0, 1) register (p = (m × 16 + 15) to (m × 16 + 15 . the value of TXQDC[3:0] bits)).
- RSCFDnCFDTMCp register (p = (m × 16) + 1, (m × 16) + 2, (m × 16) + 4, or (m × 16) + 5) corresponding to the transmit buffer allocated as a payload storage area when the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode)

Bits in the RSCFDnCFDTMCp register are all cleared to 0 in channel reset mode. Modify the RSCFDnCFDTMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCFDnCFDTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

TMTAR Bit

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.

The TMTAR bit can be set to 1 when TMTR bit is 1.

The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.

The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCFDnCFDTMSTSp register is 00b.

28.4.11.2 RSCFDnCFDTMSTSp — Transmit Buffer Status Register (p = 0 to 31)

Access Size: RSCFDnCFDTMSTSp registers can be read/written in 8-bit units

Address(es): RSCFDnCFDTMSTSp: <RSCFDn_base> + H'02D0 + (H'01 × p)

Initial Value: H'00

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
4	TMTARM	0	R	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	0	R	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	All 0	R/W	Transmit Buffer Transmit Result Status Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	0	R	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCFDnCFDTMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 1.

The TMTARM flag is set to 0 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCFDnCFDTMCp register is set to 1.

The TMTRM flag is set to 0 when the TMTR bit in the RSCFDnCFDTMCp register is set to 0.

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.

00b: Transmission is in progress or no transmit request is present.

01b: Transmission from the transmit buffer was aborted.

10b: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 0 (transmit abort is not requested).

11b: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 1 (transmit abort is requested).

Write 00b to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00b to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

28.4.11.3 RSCFDnCFDTMIDp — Transmit Buffer ID Register (p = 0 to 31)

Access Size:	RSCFDnCFDTMIDp register can be read/written in 32-bit units RSCFDnCFDTMIDpL, RSCFDnCFDTMIDpH registers can be read/written in 16-bit units RSCFDnCFDTMIDpLL, RSCFDnCFDTMIDpLH, RSCFDnCFDTMIDpHL, RSCFDnCFDTMIDpHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDTMIDp: <RSCFDn_base> + H’4000 + (H’20 × p) RSCFDnCFDTMIDpL: <RSCFDn_base> + H’4000 + (H’20 × p), RSCFDnCFDTMIDpH: <RSCFDn_base> + H’4002 + (H’20 × p) RSCFDnCFDTMIDpLL: <RSCFDn_base> + H’4000 + (H’20 × p), RSCFDnCFDTMIDpLH: <RSCFDn_base> + H’4001 + (H’20 × p), RSCFDnCFDTMIDpHL: <RSCFDn_base> + H’4002 + (H’20 × p), RSCFDnCFDTMIDpHH: <RSCFDn_base> + H’4003 + (H’20 × p)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TMIDE	0	R/W	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	0	R/W	Transmit Buffer RTR/RRS ● When the transmit message is a classical CAN frame 0: Data frame 1: Remote frame ● When the transmit message is a CANFD frame Write 0 to this bit.
29	THLEN	0	R/W	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	All 0	R/W	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

Set this bit to 0 when the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 1 (CANFD frame).

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp) of transmit messages is stored in the transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

28.4.11.4 RSCFDnCFDTMPTRp — Transmit Buffer Pointer Register (p = 0 to 31)

Access Size: RSCFDnCFDTMPTRp register can be read/written in 32-bit units
RSCFDnCFDTMPTRpL, RSCFDnCFDTMPTRpH registers can be read/written in 16-bit units
RSCFDnCFDTMPTRpLL, RSCFDnCFDTMPTRpLH, RSCFDnCFDTMPTRpHL, RSCFDnCFDTMPTRpHH registers can be read/written in 8-bit units

Address(es): RSCFDnCFDTMPTRp: <RSCFDn_base> + H'4004 + (H'20 × p)
RSCFDnCFDTMPTRpL: <RSCFDn_base> + H'4004 + H'(20 × p),
RSCFDnCFDTMPTRpH: <RSCFDn_base> + H'4006 + (H'20 × p)
RSCFDnCFDTMPTRpLL: <RSCFDn_base> + H'4004 + (H'20 × p),
RSCFDnCFDTMPTRpLH: <RSCFDn_base> + H'4005 + (H'20 × p),
RSCFDnCFDTMPTRpHL: <RSCFDn_base> + H'4006 + (H'20 × p),
RSCFDnCFDTMPTRpHH: <RSCFDn_base> + H'4007 + (H'20 × p)

Initial Value: H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31 to 28</td><td>TMDLC[3:0]</td><td>All 0</td><td>R/W</td><td>Transmit Buffer DLC Data
<table>
<tr><th>b31</th><th>b30</th><th>b29</th><th>b28</th><th>Classical CAN Frame</th><th>CANFD Frame</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0 data bytes</td><td></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1 data byte</td><td></td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>2 data bytes</td><td></td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>3 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>4 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>5 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>6 data bytes</td><td></td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>7 data bytes</td><td></td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>8 data bytes</td><td></td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>8 data bytes</td><td>12 data bytes</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td></td><td>16 data bytes</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td></td><td>20 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td></td><td>24 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td></td><td>32 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td></td><td>48 data bytes</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td></td><td>64 data bytes</td></tr>
</table></td></tr>
<tr><td>27 to 24</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
These bits are read as the value after reset. The write value should be the value after reset.</td></tr>
<tr><td>23 to 16</td><td>TMPTR[7:0]</td><td>All 0</td><td>R/W</td><td>Transmit Buffer Label Data
Set the label information to be stored in the transmit history buffer.</td></tr>
</table>

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCFDnCFDTMIDp register is set to 0 (data frame).

When the TMDLC[3:0] bits are set to 1001b or more while the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 0 (classical CAN frame), 8-byte data is transmitted actually. When the MFDF bit is 1 (CANFD frame), the settable value range varies depending on the setting of the TMME bit in the RSCFDnCFDCmFDCFG register.

- When the TMME bit = 0 (transmit buffer merge mode disabled):
  A value of 0000b to 1111b is settable. If a value larger than 1100b is set, payloads exceeding 20 bytes are padded by H’CC.
- When the TMME bit = 1 (transmit buffer merge mode enabled):
  When the corresponding transmit buffer number p = (m × 16) + 0 or (m × 16) + 3, a value of 0000b to 1111b is settable. In other cases, set a value of 0000b to 1011b (20 data bytes).

When the TMRTR bit is 1 (remote frame), set the data length of a message to be requested.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

28.4.11.5 RSCFDnCFDTMFDCTRp — Transmit Buffer CANFD Configuration Register (p = 0 to 31)

Access Size:	RSCFDnCFDTMFDCTRp register can be read/written in 32-bit units RSCFDnCFDTMFDCTRpL, RSCFDnCFDTMFDCTRpH registers can be read/written in 16-bit units RSCFDnCFDTMFDCTRpLL, RSCFDnCFDTMFDCTRpLH, RSCFDnCFDTMFDCTRpHL, RSCFDnCFDTMFDCTRpHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDTMFDCTRp: <RSCFDn_base> + H’4008 + (H’20 × p) RSCFDnCFDTMFDCTRpL: <RSCFDn_base> + H’4008 + (H’20 × p), RSCFDnCFDTMFDCTRpH: <RSCFDn_base> + H’400A + (H’20 × p) RSCFDnCFDTMFDCTRpLL: <RSCFDn_base> + H’4008 + (H’20 × p), RSCFDnCFDTMFDCTRpLH: <RSCFDn_base> + H’4009 + (H’20 × p), RSCFDnCFDTMFDCTRpHL: <RSCFDn_base> + H’400A + (H’20 × p), RSCFDnCFDTMFDCTRpHH: <RSCFDn_base> + H’400B + (H’20 × p)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TMFDF	TMBRS	TMESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	TMFDF	0	R/W	FDF 0: Classical CAN frame 1: CANFD frame
1	TMBRS	0	R/W	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	TMESI	0	R/W	ESI 0: Error active node 1: Error passive node

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is 0 (transmission not requested). When this register is linked to the transmit/receive FIFO buffer, do not write data to this register. When this register is allocated to the transmit queue, write data only to transmit buffer p (p = m × 16 + 15) of the corresponding channel.

TMFDF Bit

This bit is used to set the FD format of the message to be transmitted from the transmit buffer.

TMBRS Bit

When this bit is set to 1 while the TMFDF bit is 1, the data area of a transmit message is transmitted at the data bit rate. When the TMFDF bit is 0, write 0 to this bit.

TMESI Bit

This bit is used to set the ESI bit value of the message to be transmitted from the transmit buffer when the TMFDF bit is 1. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in the error active state. When the channel is in the error passive state, the ESI bit value that shows an error passive node is transmitted regardless of this bit value. When the TMFDF bit is 0, write 0 to this bit.

28.4.11.6 RSCFDnCFDTMDFb_p — Transmit Buffer Data Field b Register (b = 0 to 4, p = 0 to 31)

Access Size:	RSCFDnCFDTMDFb_p register can be read/written in 32-bit units RSCFDnCFDTMDFb_pL, RSCFDnCFDTMDFb_pH registers can be read/written in 16-bit units RSCFDnCFDTMDFb_pLL, RSCFDnCFDTMDFb_pLH, RSCFDnCFDTMDFb_pHL, RSCFDnCFDTMDFb_pHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDTMDFb_p: <RSCFDn_base> + H'400C + (H'04 × b) + (H'20 × p) RSCFDnCFDTMDFb_pL: <RSCFDn_base> + H'400C + (H'04 × b) + (H'20 × p), RSCFDnCFDTMDFb_pH: <RSCFDn_base> + H'400E + (H'04 × b) + (H'20 × p) RSCFDnCFDTMDFb_pLL: <RSCFDn_base> + H'400C + (H'04 × b) + (H'20 × p), RSCFDnCFDTMDFb_pLH: <RSCFDn_base> + H'400D + (H'04 × b) + (H'20 × p), RSCFDnCFDTMDFb_pHL: <RSCFDn_base> + H'400E + (H'04 × b) + (H'20 × p), RSCFDnCFDTMDFb_pHH: <RSCFDn_base> + H'400F + (H'04 × b) + (H'20 × p)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB4 × b + 3 [7:0]								TMDB4 × b + 2 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB4 × b + 1 [7:0]								TMDB4 × b + 0 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TMDB4 × b + 3 [7:0]	All 0	R/W	Transmit Buffer Data Byte 4 × b + 3 Transmit Buffer Data Byte 4 × b + 2 Transmit Buffer Data Byte 4 × b + 1 Transmit Buffer Data Byte 4 × b + 0 Set the transmit buffer data.
23 to 16	TMDB4 × b + 2 [7:0]	All 0	R/W	
15 to 8	TMDB4 × b + 1 [7:0]	All 0	R/W	
7 to 0	TMDB4 × b + 0 [7:0]	All 0	R/W	

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

28.4.11.7 RSCFDnCFDTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0)

Access Size:	RSCFDnCFDTMIECy register can be read/written in 32-bit units RSCFDnCFDTMIECyL, RSCFDnCFDTMIECyH registers can be read/written in 16-bit units RSCFDnCFDTMIECyLL, RSCFDnCFDTMIECyLH, RSCFDnCFDTMIECyHL, RSCFDnCFDTMIECyHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDTMIECy: <RSCFDn_base> + H’0390 + (H’04 × y) RSCFDnCFDTMIECyL: <RSCFDn_base> + H’0390 + (H’04 × y), RSCFDnCFDTMIECyH: <RSCFDn_base> + H’0392 + (H’04 × y) RSCFDnCFDTMIECyLL: <RSCFDn_base> + H’0390 + (H’04 × y), RSCFDnCFDTMIECyLH: <RSCFDn_base> + H’0391 + (H’04 × y), RSCFDnCFDTMIECyHL: <RSCFDn_base> + H’0392 + (H’04 × y), RSCFDnCFDTMIECyHH: <RSCFDn_base> + H’0393 + (H’04 × y)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMIEp	All 0	R/W	Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled
15 to 0	TMIEp	All 0	R/W	Transmit Buffer Interrupt Enable p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 31)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCFDnCFDTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode enable), set the bit corresponding to the transmit buffer allocated as a payload storage area to 0.

Table 28.27 shows the bit assignment.

Table 28.27 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.4.12 Details of Transmit Buffer Status-related Registers

28.4.12.1 RSCFDnCFDTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)

Access Size: RSCFDnCFDTMTRSTSy register can be read only in 32-bit units
RSCFDnCFDTMTRSTSyL, RSCFDnCFDTMTRSTSyH registers can be read only in 16-bit units
RSCFDnCFDTMTRSTSyLL, RSCFDnCFDTMTRSTSyLH, RSCFDnCFDTMTRSTSyHL, RSCFDnCFDTMTRSTSyHH registers can be read only in 8-bit units

Address(es): RSCFDnCFDTMTRSTSy: <RSCFDn_base> + H’0350 + (H’04 × y)
RSCFDnCFDTMTRSTSyL: <RSCFDn_base> + H’0350 + (H’04 × y),
RSCFDnCFDTMTRSTSyH: <RSCFDn_base> + H’0352 + (H’04 × y)
RSCFDnCFDTMTRSTSyLL: <RSCFDn_base> + H’0350 + (H’04 × y),
RSCFDnCFDTMTRSTSyLH: <RSCFDn_base> + H’0351 + (H’04 × y),
RSCFDnCFDTMTRSTSyHL: <RSCFDn_base> + H’0352 + (H’04 × y),
RSCFDnCFDTMTRSTSyHH: <RSCFDn_base> + H’0353 + (H’04 × y)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTRSTSp	All 0	R	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit request is present. 1: A transmit request is present.
15 to 0	TMTRSTSp	All 0	R	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTR bit in the RSCFDnCFDTMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 28.28 shows the bit assignment.

Table 28.28 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.4.12.2 RSCFDnCFDTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)

Access Size: RSCFDnCFDTMTARSTSy register can be read only in 32-bit units
RSCFDnCFDTMTARSTSyL, RSCFDnCFDTMTARSTSyH registers can be read only in 16-bit units
RSCFDnCFDTMTARSTSyLL, RSCFDnCFDTMTARSTSyLH, RSCFDnCFDTMTARSTSyHL, RSCFDnCFDTMTARSTSyHH registers can be read only in 8-bit units

Address(es): RSCFDnCFDTMTARSTSy: <RSCFDn_base> + H’0360 + (H’04 × y)
RSCFDnCFDTMTARSTSyL: <RSCFDn_base> + H’0360 + (H’04 × y),
RSCFDnCFDTMTARSTSyH: <RSCFDn_base> + H’0362 + (H’04 × y)
RSCFDnCFDTMTARSTSyLL: <RSCFDn_base> + H’0360 + (H’04 × y),
RSCFDnCFDTMTARSTSyLH: <RSCFDn_base> + H’0361 + (H’04 × y),
RSCFDnCFDTMTARSTSyHL: <RSCFDn_base> + H’0362 + (H’04 × y),
RSCFDnCFDTMTARSTSyHH: <RSCFDn_base> + H’0363 + (H’04 × y)

Initial Value: H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTARSTSp	All 0	R	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit abort request is present. 1: A transmit abort request is present.
15 to 0	TMTARSTSp	All 0	R	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

TMTARSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTAR bit in the RSCFDnCFDTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 28.29 shows the bit assignment.

Table 28.29 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.4.12.3 RSCFDnCFDTMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)

Access Size:	RSCFDnCFDTMTCSTSy register can be read only in 32-bit units RSCFDnCFDTMTCSTSyL, RSCFDnCFDTMTCSTSyH registers can be read only in 16-bit units RSCFDnCFDTMTCSTSyLL, RSCFDnCFDTMTCSTSyLH, RSCFDnCFDTMTCSTSyHL, RSCFDnCFDTMTCSTSyHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDTMTCSTSy: <RSCFDn_base> + H’0370 + (H’04 × y) RSCFDnCFDTMTCSTSyL: <RSCFDn_base> + H’0370 + (H’04 × y), RSCFDnCFDTMTCSTSyH: <RSCFDn_base> + H’0372 + (H’04 × y) RSCFDnCFDTMTCSTSyLL: <RSCFDn_base> + H’0370 + (H’04 × y), RSCFDnCFDTMTCSTSyLH: <RSCFDn_base> + H’0371 + (H’04 × y), RSCFDnCFDTMTCSTSyHL: <RSCFDn_base> + H’0372 + (H’04 × y), RSCFDnCFDTMTCSTSyHH: <RSCFDn_base> + H’0373 + (H’04 × y)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTCSTSp	All 0	R	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission has not been completed. 1: Transmission has been completed.
15 to 0	TMTCSTSp	All 0	R	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnCFDTMSTSp register is set to 10b (transmission has been completed (without transmit abort request)) or 11b (transmission has been completed (with transmit abort request)), the corresponding TMTCSTSp flag is set to 1.

A TMTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00b or in channel reset mode.

Table 28.30 shows the bit assignment.

Table 28.30 TMTCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.4.12.4 RSCFDnCFDTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0)

Access Size: RSCFDnCFDTMTASTSy register can be read only in 32-bit units
RSCFDnCFDTMTASTSyL, RSCFDnCFDTMTASTSyH registers can be read only in 16-bit units
RSCFDnCFDTMTASTSyLL, RSCFDnCFDTMTASTSyLH, RSCFDnCFDTMTASTSyHL, RSCFDnCFDTMTASTSyHH registers can be read only in 8-bit units

Address(es): RSCFDnCFDTMTASTSy: <RSCFDn_base> + H’0380 + (H’04 × y)
RSCFDnCFDTMTASTSyL: <RSCFDn_base> + H’0380 + (H’04 × y),
RSCFDnCFDTMTASTSyH: <RSCFDn_base> + H’0382 + (H’04 × y)
RSCFDnCFDTMTASTSyLL: <RSCFDn_base> + H’0380 + (H’04 × y),
RSCFDnCFDTMTASTSyLH: <RSCFDn_base> + H’0381 + (H’04 × y),
RSCFDnCFDTMTASTSyHL: <RSCFDn_base> + H’0382 + (H’04 × y),
RSCFDnCFDTMTASTSyHH: <RSCFDn_base> + H’0383 + (H’04 × y)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31 to y x 32 + 16 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0))															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTASTSp	All 0	R	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted 1: Transmission is aborted
15 to 0	TMTASTSp	All 0	R	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnCFDTMSTSp register is set to 01b (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00b or in channel reset mode.

Table 28.31 shows the bit assignment.

Table 28.31 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15

28.4.13 Details of Transmit Queue-related Registers

28.4.13.1 RSCFDnCFDTXQCCm — Transmit Queue Configuration and Control Register (m = 0, 1)

Access Size:	RSCFDnCFDTXQCCm register can be read/written in 32-bit units RSCFDnCFDTXQCCmL, RSCFDnCFDTXQCCmH registers can be read/written in 16-bit units RSCFDnCFDTXQCCmLL, RSCFDnCFDTXQCCmLH, RSCFDnCFDTXQCCmHL, RSCFDnCFDTXQCCmHH registers can be read/ written in 8-bit units
Address(es):	RSCFDnCFDTXQCCm: <RSCFDn_base> + H'03A0 + (H'04 × m) RSCFDnCFDTXQCCmL: <RSCFDn_base> + H'03A0 + (H'04 × m), RSCFDnCFDTXQCCmH: <RSCFDn_base> + H'03A2 + (H'04 × m) RSCFDnCFDTXQCCmLL: <RSCFDn_base> + H'03A0 + (H'04 × m), RSCFDnCFDTXQCCmLH: <RSCFDn_base> + H'03A1 + (H'04 × m), RSCFDnCFDTXQCCmHL: <RSCFDn_base> + H'03A2 + (H'04 × m), RSCFDnCFDTXQCCmHH: <RSCFDn_base> + H'03A3 + (H'04 × m)
Initial Value:	H'0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
13	TXQIM	0	R/W	Transmit Queue Interrupt Source Select 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	0	R/W	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	All 0	R/W	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited. For transmit buffer merge mode, set g to 2 to 9.
7 to 1	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
0	TXQE	0	R/W	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.

Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see **Figure 28.9**.

When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode), transmit buffers (m × 16 + 5) to (m × 16 + 0) are merged and cannot be allocated to the transmit queue. Therefore, do not set TXQDC[3:0] bits to 10 to 15.

Modify these bits only in channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to 0010b or more.

28.4.13.2 RSCFDnCFDTXQSTSm — Transmit Queue Status Register (m = 0, 1)

Access Size:	RSCFDnCFDTXQSTSm register can be read/written in 32-bit units RSCFDnCFDTXQSTSmL, RSCFDnCFDTXQSTSmH registers can be read/written in 16-bit units RSCFDnCFDTXQSTSmLL, RSCFDnCFDTXQSTSmLH, RSCFDnCFDTXQSTSmHL, RSCFDnCFDTXQSTSmHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDTXQSTSm: <RSCFDn_base> + H’03C0 + (H’04 × m) RSCFDnCFDTXQSTSmL: <RSCFDn_base> + H’03C0 + (H’04 × m), RSCFDnCFDTXQSTSmH: <RSCFDn_base> + H’03C2 + (H’04 × m) RSCFDnCFDTXQSTSmLL: <RSCFDn_base> + H’03C0 + (H’04 × m), RSCFDnCFDTXQSTSmLH: <RSCFDn_base> + H’03C1 + (H’04 × m), RSCFDnCFDTXQSTSmHL: <RSCFDn_base> + H’03C2 + (H’04 × m), RSCFDnCFDTXQSTSmHH: <RSCFDn_base> + H’03C3 + (H’04 × m)
Initial Value:	H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
12 to 8	—	All 0	R	Reserved When read, the undefined value is returned. When writing to these bits, write the value after reset.
7 to 3	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	TXQIF	0	R/W*1	Transmit Queue Interrupt Request Flag 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.
1	TXQFLL	0	R	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	1	R	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCFDnCFDTXQCCm register has occurred.

The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCFDnCFDTXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCFDnCFDTXQCCm register.

This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.

This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

28.4.13.3 RSCFDnCFDTXQPCTRm — Transmit Queue Pointer Control Register (m = 0, 1)

Access Size: RSCFDnCFDTXQPCTRm register can only be written in 32-bit units
RSCFDnCFDTXQPCTRmL, RSCFDnCFDTXQPCTRmH registers can only be written in 16-bit units
RSCFDnCFDTXQPCTRmLL, RSCFDnCFDTXQPCTRmLH, RSCFDnCFDTXQPCTRmHL, RSCFDnCFDTXQPCTRmHH registers can only be written in 8-bit units

Address(es): RSCFDnCFDTXQPCTRm: <RSCFDn_base> + H’03E0 + (H’04 × m)
RSCFDnCFDTXQPCTRmL: <RSCFDn_base> + H’03E0 + (H’04 × m),
RSCFDnCFDTXQPCTRmH: <RSCFDn_base> + H’03E2 + (H’04 × m)
RSCFDnCFDTXQPCTRmLL: <RSCFDn_base> + H’03E0 + (H’04 × m),
RSCFDnCFDTXQPCTRmLH: <RSCFDn_base> + H’03E1 + (H’04 × m),
RSCFDnCFDTXQPCTRmHL: <RSCFDn_base> + H’03E2 + (H’04 × m),
RSCFDnCFDTXQPCTRmHH: <RSCFDn_base> + H’03E3 + (H’04 × m)

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should be the value after reset.
7 to 0	TXQPC[7:0]	All 0	W	Transmit Queue Pointer Control Writing H’FF to these bits moves the write pointer of the transmit queue to the next queue buffer.

TXQPC[7:0] Bits

Writing H’FF to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCFDnCFDTMIDp, RSCFDnCFDTMPTRp, RSCFDnCFDTMFDCTRp, and RSCFDnCFDTMDFb_p registers (p = 15 and 31) before writing H’FF to the TXQPC[7:0] bits.

When writing H’FF to these bits, make sure that the TXQE bit in the RSCFDnCFDTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCFDnCFDTXQSTSm register is 0 (he transmit queue is not full).

28.4.14 Details of Transmit History-related Registers

28.4.14.1 RSCFDnCFDTHLCCm — Transmit History Configuration and Control Register (m = 0, 1)

Access Size:	RSCFDnCFDTHLCCm register can be read/written in 32-bit units RSCFDnCFDTHLCCmL, RSCFDnCFDTHLCCmH registers can be read/written in 16-bit units RSCFDnCFDTHLCCmLL, RSCFDnCFDTHLCCmLH, RSCFDnCFDTHLCCmHL, RSCFDnCFDTHLCCmHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDTHLCCm: <RSCFDn_base> + H’0400 + (H’04 × m) RSCFDnCFDTHLCCmL: <RSCFDn_base> + H’0400 + (H’04 × m), RSCFDnCFDTHLCCmH: <RSCFDn_base> + H’0402 + (H’04 × m) RSCFDnCFDTHLCCmLL: <RSCFDn_base> + H’0400 + (H’04 × m), RSCFDnCFDTHLCCmLH: <RSCFDn_base> + H’0401 + (H’04 × m), RSCFDnCFDTHLCCmHL: <RSCFDn_base> + H’0402 + (H’04 × m), RSCFDnCFDTHLCCmHH: <RSCFDn_base> + H’0403 + (H’04 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
10	THLDTE	0	R/W	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	0	R/W	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	0	R/W	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	—	All 0	R	Reserved When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	THLE	0	R/W	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.

Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer.

Modify this bit in channel communication mode or channel halt mode.

This bit is cleared to 0 in channel reset mode.

28.4.14.2 RSCFDnCFDTHLSTSm — Transmit History Status Register (m = 0, 1)

Access Size:	RSCFDnCFDTHLSTSm register can be read/written in 32-bit units RSCFDnCFDTHLSTSmL, RSCFDnCFDTHLSTSmH registers can be read/written in 16-bit units RSCFDnCFDTHLSTSmLL, RSCFDnCFDTHLSTSmLH, RSCFDnCFDTHLSTSmHL, RSCFDnCFDTHLSTSmHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDTHLSTSm: <RSCFDn_base> + H’0420 + (H’04 × m) RSCFDnCFDTHLSTSmL: <RSCFDn_base> + H’0420 + (H’04 × m), RSCFDnCFDTHLSTSmH: <RSCFDn_base> + H’0422 + (H’04 × m) RSCFDnCFDTHLSTSmLL: <RSCFDn_base> + H’0420 + (H’04 × m), RSCFDnCFDTHLSTSmLH: <RSCFDn_base> + H’0421 + (H’04 × m), RSCFDnCFDTHLSTSmHL: <RSCFDn_base> + H’0422 + (H’04 × m), RSCFDnCFDTHLSTSmHH: <RSCFDn_base> + H’0423 + (H’04 × m)
Initial Value:	H’0000 0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THLELT	THLFLL	THLEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
12 to 8	THLMC[4:0]	All 0	R	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
3	THLIF	0	R/W*1	Transmit History Interrupt Request Flag 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present.
2	THLELT	0	R/W*1	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	0	R	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	1	R	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state

THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer. These bits are cleared to 0 in channel reset mode.

THLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCFDnCFDTHLCCm register occurs.

This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.

When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLELT Flag

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.

When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.

This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).

NOTE

To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

28.4.14.3 RSCFDnCFDTHLPCTRm — Transmit History Pointer Control Register (m = 0, 1)

Access Size:	RSCFDnCFDTHLPCTRm register can only be written in 32-bit units RSCFDnCFDTHLPCTRmL, RSCFDnCFDTHLPCTRmH registers can only be written in 16-bit units RSCFDnCFDTHLPCTRmLL, RSCFDnCFDTHLPCTRmLH, RSCFDnCFDTHLPCTRmHL, RSCFDnCFDTHLPCTRmHH registers can only be written in 8-bit units
Address(es):	RSCFDnCFDTHLPCTRm: <RSCFDn_base> + H’0440 + (H’04 × m) RSCFDnCFDTHLPCTRmL: <RSCFDn_base> + H’0440 + (H’04 × m), RSCFDnCFDTHLPCTRmH: <RSCFDn_base> + H’0442 + H’(04 × m) RSCFDnCFDTHLPCTRmLL: <RSCFDn_base> + H’0440 + (H’04 × m), RSCFDnCFDTHLPCTRmLH: <RSCFDn_base> + H’0441 + (H’04 × m), RSCFDnCFDTHLPCTRmHL: <RSCFDn_base> + H’0442 + (H’04 × m), RSCFDnCFDTHLPCTRmHH: <RSCFDn_base> + H’0443 + (H’04 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved When writing to these bits, write the value after reset.
7 to 0	THLPC[7:0]	All 0	W	Transmit History List Pointer Control Writing H’FF to these bits moves the read pointer to the next unread data in the transmit history buffer.

THLPC[7:0] Bits

When the THLPC[7:0] bits are set to H’FF, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCFDnCFDTHLSTSm register is decremented. Write H’FF to the THLPC[7:0] bits after reading from the RSCFDnCFDTHLACCm register.

When writing H’FF to these bits, make sure that the THLE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCFDnCFDTHLSTSm register is 0.

28.4.14.4 RSCFDnCFDTHLACCm — Transmit History Access Register (m = 0, 1)

Access Size:	RSCFDnCFDTHLACCm register can be read only in 32-bit units RSCFDnCFDTHLACCmL, RSCFDnCFDTHLACCmH registers can be read only in 16-bit units RSCFDnCFDTHLACCmLL, RSCFDnCFDTHLACCmLH, RSCFDnCFDTHLACCmHL, RSCFDnCFDTHLACCmHH registers can be read only in 8-bit units
Address(es):	RSCFDnCFDTHLACCm: <RSCFDn_base> + H’6000 + (H’04 × m) RSCFDnCFDTHLACCmL: <RSCFDn_base> + H’6000 + (H’04 × m), RSCFDnCFDTHLACCmH: <RSCFDn_base> + H’6002 + (H’04 × m) RSCFDnCFDTHLACCmLL: <RSCFDn_base> + H’6000 + (H’04 × m), RSCFDnCFDTHLACCmLH: <RSCFDn_base> + H’6001 + (H’04 × m), RSCFDnCFDTHLACCmHL: <RSCFDn_base> + H’6002 + (H’04 × m), RSCFDnCFDTHLACCmHH: <RSCFDn_base> + H’6003 + (H’04 × m)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTS[15:0]	All 0	R	Timestamp Data The timestamp data of stored data can be read.
15 to 8	TID[7:0]	All 0	R	Label Data The label information of stored data can be read.
7	—	All 0	R	Reserved This bit is read as the value after reset.
6 to 3	BN[3:0]	All 0	R	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	All 0	R	Buffer Type Data b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue

TMTS[15:0] Bits

Timestamp values in transmit history data stored in the transmit history buffer are displayed.

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer.

28.4.15 Details of Test-related Registers

28.4.15.1 RSCFDnCFDGTSTCFG — Global Test Configuration Register

Access Size:	RSCFDnCFDGTSTCFG register can be read/written in 32-bit units RSCFDnCFDGTSTCFGL, RSCFDnCFDGTSTCFGH registers can be read/written in 16-bit units RSCFDnCFDGTSTCFGLL, RSCFDnCFDGTSTCFGLH, RSCFDnCFDGTSTCFGHL, RSCFDnCFDGTSTCFGHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGTSTCFG: <RSCFDn_base> + H’0468 RSCFDnCFDGTSTCFGL: <RSCFDn_base> + H’0468, RSCFDnCFDGTSTCFGH: <RSCFDn_base> + H’046A RSCFDnCFDGTSTCFGLL: <RSCFDn_base> + H’0468, RSCFDnCFDGTSTCFGLH: <RSCFDn_base> + H’0469, RSCFDnCFDGTSTCFGHL: <RSCFDn_base> + H’046A, RSCFDnCFDGTSTCFGHH: <RSCFDn_base> + H’046B
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	C1ICBCE	C0ICBCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
22 to 16	RTMPS[6:0]	All 0	R/W	RAM Test Page Configuration Set a value within a range of page 0 (H’00) to page 27 (H’1B).
15 to 2	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
1	C1ICBCE	0	R/W	CAN1 Inter-Channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	0	R/W	CAN0 Inter-Channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCFDnCFDGTSTCFG register only in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of H’00 to H’1B, inclusive. Do not access the RAM area higher than the 192th byte on the last page (the setting of the RTMPS bit is H’1B).

C1ICBCE Bit

Setting this bit to 1 enables the channel 1 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C0ICBCE Bit

Setting this bit to 1 enables the channel 0 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

28.4.15.2 RSCFDnCFDGTSTCTR — Global Test Control Register

Access Size:	RSCFDnCFDGTSTCTR register can be read/written in 32-bit units RSCFDnCFDGTSTCTRL, RSCFDnCFDGTSTCTRH registers can be read/written in 16-bit units RSCFDnCFDGTSTCTRLL, RSCFDnCFDGTSTCTRLH, RSCFDnCFDGTSTCTRHL, RSCFDnCFDGTSTCTRHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDGTSTCTR: <RSCFDn_base> + H’046C RSCFDnCFDGTSTCTRL: <RSCFDn_base> + H’046C, RSCFDnCFDGTSTCTRH: <RSCFDn_base> + H’046E RSCFDnCFDGTSTCTRLL: <RSCFDn_base> + H’046C, RSCFDnCFDGTSTCTRLH: <RSCFDn_base> + H’046D, RSCFDnCFDGTSTCTRHL: <RSCFDn_base> + H’046E, RSCFDnCFDGTSTCTRHH: <RSCFDn_base> + H’046F
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are read as the value after reset. The write value should be the value after reset.
2	RTME	0	R/W	RAM Test Enable 0: RAM test is disabled. 1: RAM test is enabled.
1	—	0	R	Reserved This bit is read as the value after reset. The write value should be the value after reset.
0	ICBCTME	0	R/W	Communication Test between Channels Enable 0: Communication test between channels disabled 1: Communication test between channels enabled

RTME Bit

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode. This bit is cleared to 0 in global reset mode.

1. Set the GMDC[1:0] bits in the RSCFDnCFDGCTR register to 10b (Global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0, 1) in the RSCFDnCFDGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode.

This bit is cleared to 0 in global reset mode.

28.4.15.3 RSCFDnCFDGLOCKK — Global Lock Key Register

Access Size: RSCFDnCFDGLOCKK register can only be written in 32-bit units
RSCFDnCFDGLOCKKL, RSCFDnCFDGLOCKKH registers can only be written in 16-bit units

Address(es): RSCFDnCFDGLOCKK: <RSCFDn_base> + H’047C
RSCFDnCFDGLOCKKL: <RSCFDn_base> + H’047C, RSCFDnCFDGLOCKKH: <RSCFDn_base> + H’047E

Initial Value: H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved When writing these bits, write the value after reset.
15 to 0	LOCK[15:0]	All 0	W*1	Lock Key These bits are key bits to release protection of test mode.

Note 1. Writing to these bits is effective only when the RS-CANFD module is in global test mode.

The RSCFDnCFDGLOCKK register releases protection of special test bits and is write only.

For the protection release data, see **Section 28.11.4.2, Procedure for Releasing the Protection**.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCFDnCFDGTSTCTR register.

After the protection has been released, writing to the I/O register area (<RSCFDn_base> + H’0000 to <RSCFDn_base> + H’05FF) of the CAN (except the RAM) enables the protection again.

Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.

28.4.15.4 RSCFDnCFDRPGACCr — RAM Test Page Access Register (r = 0 to 63)

Access Size:	RSCFDnCFDRPGACCr register can be read/written in 32-bit units RSCFDnCFDRPGACCrL, RSCFDnCFDRPGACCrH registers can be read/written in 16-bit units RSCFDnCFDRPGACCrLL, RSCFDnCFDRPGACCrLH, RSCFDnCFDRPGACCrHL, RSCFDnCFDRPGACCrHH registers can be read/written in 8-bit units
Address(es):	RSCFDnCFDRPGACCr: <RSCFDn_base> + H’6400 + (H’04 × r) RSCFDnCFDRPGACCrL: <RSCFDn_base> + H’6400 + (H’04 × r), RSCFDnCFDRPGACCrH: <RSCFDn_base> + H’6402 + (H’04 × r) RSCFDnCFDRPGACCrLL: <RSCFDn_base> + H’6400 + (H’04 × r), RSCFDnCFDRPGACCrLH: <RSCFDn_base> + H’6401 + (H’04 × r), RSCFDnCFDRPGACCrHL: <RSCFDn_base> + H’6402 + (H’04 × r), RSCFDnCFDRPGACCrHH: <RSCFDn_base> + H’6403 + (H’04 × r)
Initial Value:	H’0000 0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDTA[31:0]	All 0	R/W	RAM Data Test Access Data can be read and written in RSCAN RAM.

Modify the RSCFDnCFDRPGACCr register in global test mode with the RTME bit in the RSCFDnCFDGTSTCTR register set to 1 (RAM test is enabled).

The RSCFDnCFDRPGACCr register is readable and writable when the RTME bit is set to 1.

28.5 Interrupt Sources and DMA Trigger

28.5.1 Interrupt Sources

The RS-CANFD module has 8 interrupts that are grouped into global interrupts and channel interrupts.

- Global interrupts (2 sources):

  Receive FIFO interrupt

  Global error interrupt

- Channel interrupts (3 sources/channel):

  CANm transmit interrupt (m = 0, 1)

  - CANm transmit complete interrupt
  - CANm transmit abort interrupt
  - CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode)
  - CANm transmit history interrupt
  - CANm transmit queue Interrupt

  CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode)

  CANm error interrupt

When an interrupt request is generated, the corresponding interrupt request flag is set to 1 (interrupt request present). In that case, when the interrupt enable bit is set to 1 (enabling interrupts), an interrupt request is output from the RS-CANFD module. (Generation of interrupts also depends on the interrupt control register settings of the interrupt controller.)

Setting the interrupt request flag to 0 (no interrupt request present) or setting the interrupt enable bit to 0 (disabling interrupts) clears the current interrupt request. The current interrupt request is still output until the interrupt request flag is cleared.

Table 28.32 lists the CAN interrupt sources. **Figure 28.2** shows the CAN global interrupt block diagram. **Figure 28.3** shows the CAN channel interrupt block diagram.

Table 28.32 List of CAN Interrupt Sources (1/2)

RCAN					INTC	DMA
	Interrupt Source (Unit Interrupt Signal)		Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit	Request Source Name	Transfer Request Signal
Global interrupts	Receive FIFO (INTRCANGRECC)	Receive FIFO 0	RFIF in the RSCFDn(CFD)RFSTS0 register	RFIE in the RSCFDn(CFD)RFCC0 register	RFI	RXF_DMA0
		Receive FIFO 1	RFIF in the RSCFDn(CFD)RFSTS1 register	RFIE in the RSCFDn(CFD)RFCC1 register		RXF_DMA1
		Receive FIFO 2	RFIF in the RSCFDn(CFD)RFSTS2 register	RFIE in the RSCFDn(CFD)RFCC2 register		RXF_DMA2
		Receive FIFO 3	RFIF in the RSCFDn(CFD)RFSTS3 register	RFIE in the RSCFDn(CFD)RFCC3 register		RXF_DMA3
		Receive FIFO 4	RFIF in the RSCFDn(CFD)RFSTS4 register	RFIE in the RSCFDn(CFD)RFCC4 register		RXF_DMA4
		Receive FIFO 5	RFIF in the RSCFDn(CFD)RFSTS5 register	RFIE in the RSCFDn(CFD)RFCC5 register		RXF_DMA5
		Receive FIFO 6	RFIF in the RSCFDn(CFD)RFSTS6 register	RFIE in the RSCFDn(CFD)RFCC6 register		RXF_DMA6
		Receive FIFO 7	RFIF in the RSCFDn(CFD)RFSTS7 register	RFIE in the RSCFDn(CFD)RFCC7 register		RXF_DMA7
Global interrupts	Global error (INTRCANGERR)		● DEF in the RSCFDn(CFD)GERFL register ● MES in the RSCFDn(CFD)GERFL register ● THLES in the RSCFDn(CFD)GERFL register ● CMPOF in the RSCFDnCFDGERFL register	● DEIE in the RSCFDn(CFD)GCTR register ● MEIE in the RSCFDn(CFD)GCTR register ● THLEIE in the RSCFDn(CFD)GCTR register ● CMPOFIE in the RSCFDnCFDGCTR register	GERI	—
Channel interrupts (m = 0, 1)	CANm transmit (INTRCANmTRX)	CANm transmit complete	TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register	TMIEp in the RSCFDn(CFD)TMIECy register	CTXIm	—
		CANm transmit abort	TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register	TAIE in the RSCFDn(CFD)CmCTR register		—
		CANm transmit/ receive FIFO transmit complete	CFTXIF in the RSCFDn(CFD)CFSTSk register	CFTXIE in the RSCFDn(CFD)CFCCk register		—
		CANm transmit queue	TXQIF in the RSCFDn(CFD)TXQSTSm register	TXQIE in the RSCFDn(CFD)TXQCCm register		—
		CANm transmit history	THLIF in the RSCFDn(CFD)THLSTSm register	THLIE in the RSCFDn(CFD)THLCCm register		—
	CANm transmit/receive FIFO receive complete (INTRCANmREC)		CFRXIF in the RSCFDn(CFD)CFSTSk register	CFRXIE in the RSCFDn(CFD)CFCCk register	CFRXIm	COM_DMAm

Table 28.32 List of CAN Interrupt Sources (2/2)

RCAN				INTC	DMA
	Interrupt Source (Unit Interrupt Signal)	Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit	Request Source Name	Transfer Request Signal
Channel interrupts (m = 0, 1)	CANm error (INTRCANmERR)	• BEF in the RSCFDn(CFD)CmERFL register • ALF in the RSCFDn(CFD)CmERFL register • BLF in the RSCFDn(CFD)CmERFL register • OVLF in the RSCFDn(CFD)CmERFL register • BORF in the RSCFDn(CFD)CmERFL register • BOEF in the RSCFDn(CFD)CmERFL register • EPF in the RSCFDn(CFD)CmERFL register • EWF in the RSCFDn(CFD)CmERFL register • SOCO in the RSCFDnCFDCmFDSTS register • EOCO in the RSCFDnCFDCmFDSTS register • TDCVF in the RSCFDnCFDCmFDSTS register	• BEIE in the RSCFDn(CFD)CmCTR register • ALIE in the RSCFDn(CFD)CmCTR register • BLIE in the RSCFDn(CFD)CmCTR register • OLIE in the RSCFDn(CFD)CmCTR register • BORIE in the RSCFDn(CFD)CmCTR register • BOEIE in the RSCFDn(CFD)CmCTR register • EPIE in the RSCFDn(CFD)CmCTR register • EWIE in the RSCFDn(CFD)CmCTR register • SOCOIE in the RSCFDnCFDCmCTR register • EOCOIE in the RSCFDnCFDCmCTR register • TDCVFIE in the RSCFDnCFDCmCTR register	CERIm	—

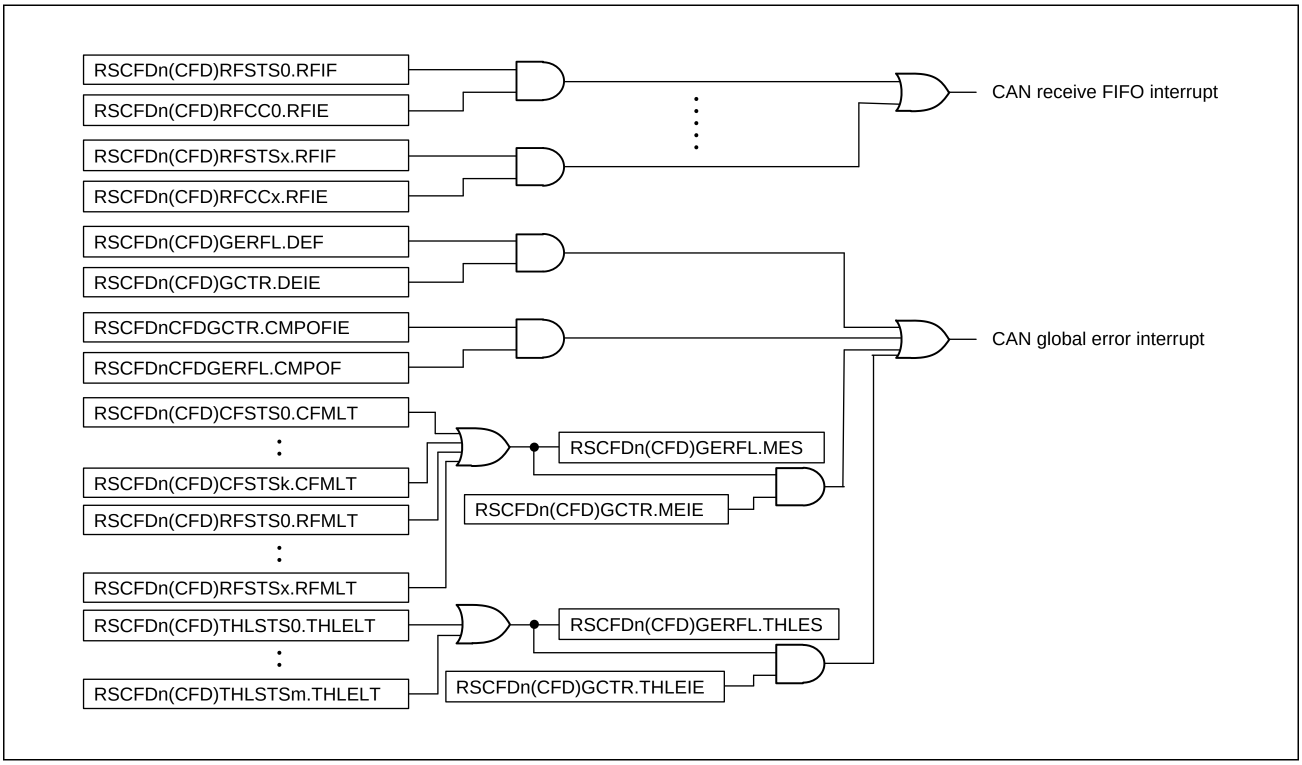


Figure 28.2 CAN Global Interrupt Block Diagram

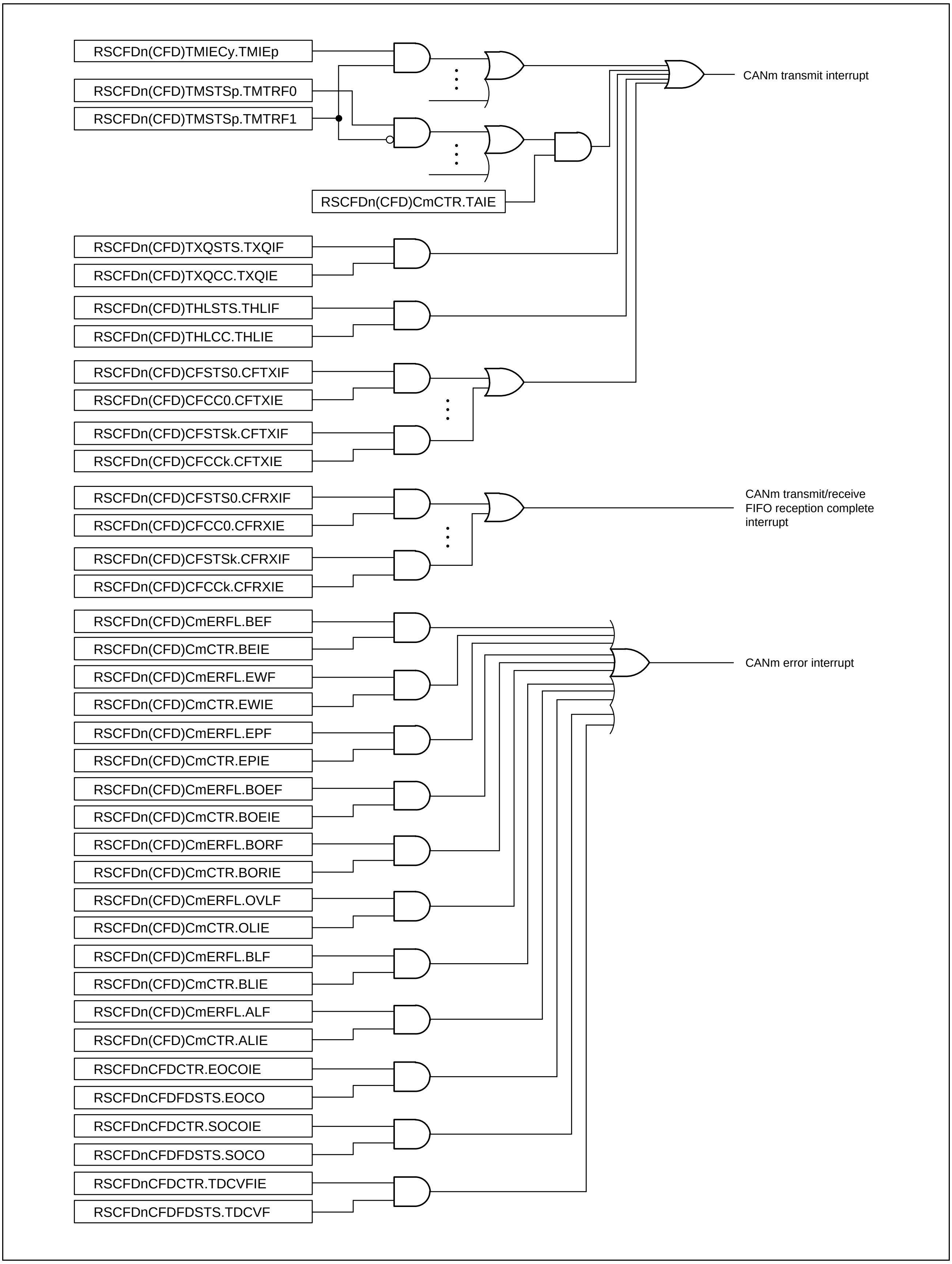


Figure 28.3 CAN Channel Interrupt Block Diagram

28.5.2 DMA Trigger (Only in CANFD Mode)

In CANFD mode, receive FIFO buffers can be related to DMA channels. The following 10 FIFO buffers can be related.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k (k = 3 × m, m = 0, 1) allocated to channel m

When the DMA enable bit (RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTCT register) is set to 1 and an unread message is remaining in the related FIFO, a DMA transfer request trigger is generated.

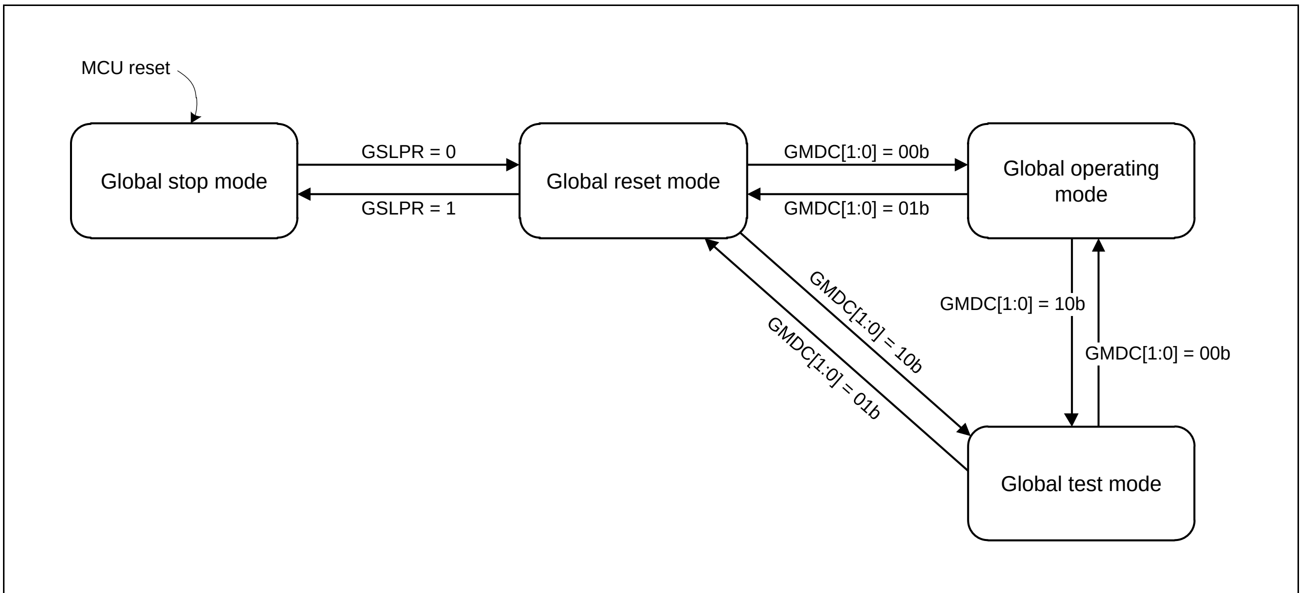
28.6 CAN Modes

The RS-CANFD module has four global modes to control the entire RS-CANFD module status and four channel modes to control individual channel status. Details of global modes are described in **Section 28.6.1, Global Modes**, and details of channel modes are described in **Section 28.6.2**, **Figure 28.5, Channel Mode State Transition Chart**.

- Global stop mode:Stops the clocks of the entire module to achieve low power consumption.
- Global reset mode:Performs initial settings for the entire module.
- Global test mode:Performs test settings and performs the RAM test.
- Global operating mode:Makes the entire module operable.
- Channel stop mode:Stops the channel clock.
- Channel reset mode:Performs initial settings for the channels.
- Channel halt mode:Stops CAN communication and allows channel testing.
- Channel communication mode:Performs CAN communication.

28.6.1 Global Modes

Figure 28.4 shows the transitions of global modes.



Note: GSLPR, GMDC[1:0]: Bits in the RSCFDn(CFD)GCTR register

Figure 28.4 Transitions of Global Modes

In some cases, global mode transitions also force channel mode transitions. **Table 28.33** shows the channel mode transitions depending on the global mode setting dictated by the GMDC[1:0] bits and the GSLPR bit.

Table 28.33 Transitions of Channel Modes Depending on Global Mode Setting (GMDC[1:0] and GSLPR Bits)

Channel Mode before Setting	Channel Mode after Setting			
	GMDC[1:0] = 00b GSLPR = 0 (Global Operation)	GMDC[1:0] = 10b GSLPR = 0 (Global Test)	GMDC[1:0] = 01b GSLPR = 0 (Global Reset)	GMDC[1:0] = 01b GSLPR = 1 (Global Stop)
Channel communication	Channel communication	Channel halt	Channel reset	Transition prohibited
Channel halt	Channel halt	Channel halt	Channel reset	Transition prohibited
Channel reset	Channel reset	Channel reset	Channel reset	Channel stop
Channel stop	Channel stop	Channel stop	Channel stop	Channel stop

Note: GMDC[1:0], GSLPR: Bits in the RSCFDn(CFD)GCTR register

Table 28.34 shows the global mode transition time.

Table 28.34 Global Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Global stop	Global reset	Three pclk cycles
Global reset	Global stop	Three pclk cycles
Global reset	Global test	Ten pclk cycles
Global reset	Global operating	Ten pclk cycles
Global test	Global reset	Two CAN bit times*1 *2
Global test	Global operating	Three pclk cycles
Global operating	Global reset	Two CAN bit times*1 *2
Global operating	Global test	Two CAN frames*1

Note 1. CAN frame time and CAN bit time of the lowest communication speed of the channels in use

Note 2. In CANFD mode, this time value is the CAN bit time of the nominal bit rate.

28.6.1.1 Global Stop Mode

In global stop mode, clocks of the CAN do not run and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained. Only the clock used by the CPU for writing to the GSLPR bit runs in this mode.

After the MCU is reset, the CAN module transitions to global stop mode. Setting the GSLPR bit in the RSCFDn(CFD)GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in each of the RSCFDn(CFD)CmCTR register to 1 (channel stop mode). Afterwards, if all channels are forced to transition to channel stop mode, the CAN module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode or global test mode.

28.6.1.2 Global Reset Mode

In global reset mode, RS-CANFD module settings are performed. When the RS-CANFD module transitions to global reset mode, some registers are initialized. For registers to be initialized, see **Table 28.37, Registers Initialized in Global Reset Mode or Channel Reset Mode** and **Table 28.38, Registers Initialized Only in Global Reset Mode**.

Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to 01b sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR registers (m = 0, 1) to 01b (channel reset mode). If all channels are forced to transition to channel reset mode, the CAN module transitions to global reset mode. Channels that are already in channel reset mode or channel stop mode do not transition (because the CHMDC[1:0] bits have already been set to 01b).

28.6.1.3 Global Test Mode

In global test mode, settings for test-related registers are performed. When the CAN module transitions to global test mode, all CAN communications are disabled.

Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to 10b sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR register to 10b (channel halt mode). If all channels are forced to transition to channel halt mode, the CAN module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

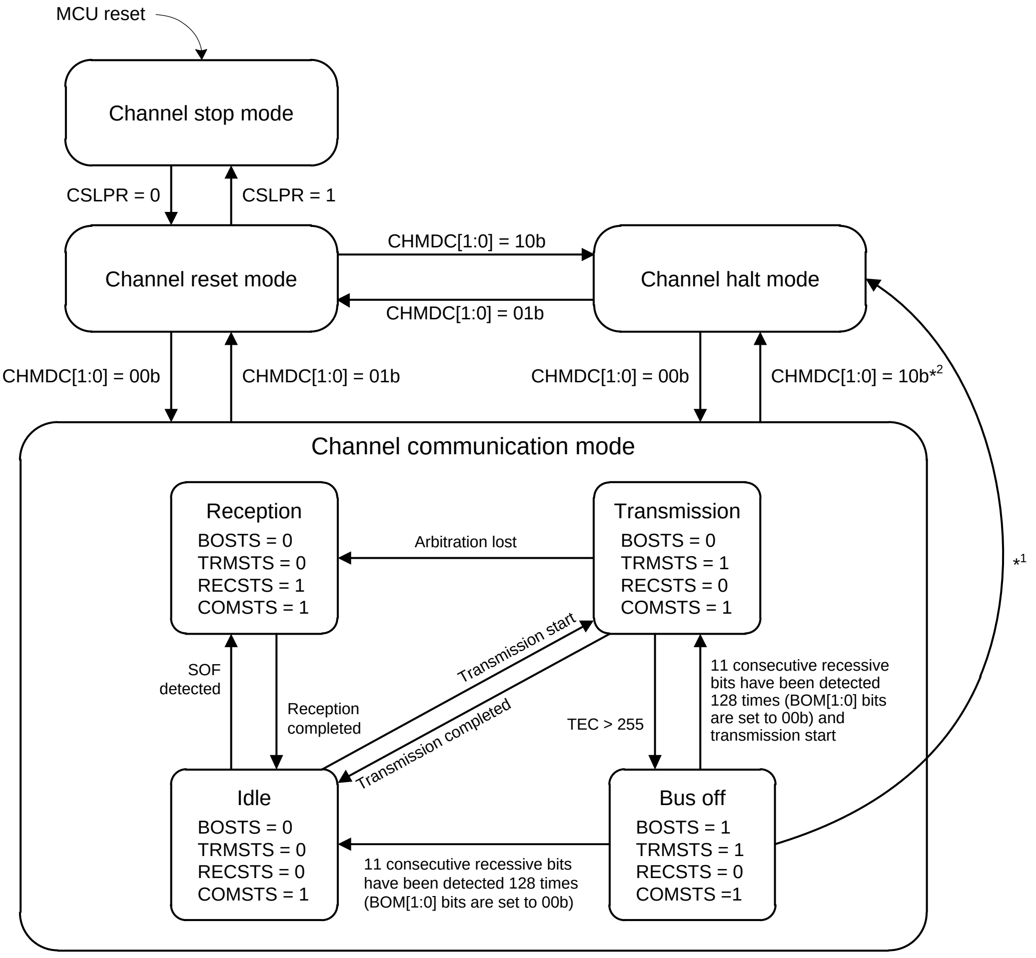
28.6.1.4 Global Operating Mode

The RS-CANFD module operates in global operating mode.

When the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register are set to 00b, the RS-CANFD module transitions to global operating mode.

28.6.2 Channel Modes

Figure 28.5 shows a channel mode state transition chart. **Table 28.35** shows the channel mode transition time.



Note: CHMDC[1:0], CSLPR, BOM[1:0]: Bits in the RSCFDn(CFD)CmCTR register (m = 0 to 1) BOSTS, TRMSTS, RECSTS, COMSTS: Bits in the RSCFDn(CFD)CmSTS register

Note 1. Timing of transition from bus off state to channel halt mode
- When BOM[1:0] = 01b: Transition to channel halt mode when TEC exceeds 255
- When BOM[1:0] = 10b: Transition to channel halt mode when 11 consecutive recessive bits have been detected 128 times
- When BOM[1:0] = 11b: Transition to channel halt mode when the CHMDC[1:0] bits are set to 10b

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode.

Figure 28.5 Channel Mode State Transition Chart

Table 28.35 Channel Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Channel stop	Channel reset	Three pclk cycles
Channel reset	Channel stop	Three pclk cycles
Channel reset	Channel halt	Three CANm bit times*[1]
Channel reset	Channel communication	Four CANm bit times*[1]
Channel halt	Channel reset	Two CANm bit times*[1]
Channel halt	Channel communication	Four CANm bit times*[1]
Channel communication	Channel reset	Two CANm bit times*[1]
Channel communication	Channel halt	Two CANm frames

Note 1. In CANFD mode, this time value is the CANm bit time of the nominal bit rate.

28.6.2.1 Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained.

Each channel enters channel stop mode after the MCU is reset. Channels also transition to channel stop mode when the GSLPR bit in the RSCFDn(CFD)CmCTR register (m = 0, 1) is set to 1 (channel stop mode) in channel reset mode. The GSLPR bit should not be modified in channel communication mode and channel halt mode.

28.6.2.2 Channel Reset Mode

In channel reset mode, channel settings are performed. When a channel transitions to channel reset mode, some channel- related registers are initialized. For registers to be initialized, see **Table 28.37, Registers Initialized in Global Reset Mode or Channel Reset Mode**.

When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 01b (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. **Table 28.36** shows the operation when the CHMDC[1:0] bits are set to 01b (channel reset mode) during CAN communication.

28.6.2.3 Channel Halt Mode

In channel halt mode, settings for test-related registers of channels are performed. When a channel transitions to channel halt mode, CAN communication of the channel stops.

Table 28.36 shows operation when the CHMDC[1:0] bits are set to 10b (channel halt mode) during CAN communication.

Table 28.36 Operation in Transitions to Channel Reset Mode/Channel Halt Mode

Mode	During Reception	During Transmission	Bus Off State
Channel reset (CHMDC[1:0] = 01b)	Transitions to channel reset mode before reception is completed.*[1]	Transitions to channel reset mode before transmission is completed.*[1]	Transitions to channel reset mode before bus off recovery.
Channel halt*[3] (CHMDC[1:0] = 10b)	Transitions to channel halt mode after reception is completed.*[2]	Transitions to channel halt mode after transmission is completed.	[When BOM[1:0] = 00b] Transitions to channel halt mode (CHMDC[1:0] = 10b) only after bus off recovery. [When BOM[1:0] = 01b] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = 10b] Transitions to channel halt mode automatically after bus off recovery. [When BOM[1:0] = 11b] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to 10b before bus off recovery.

Note 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to 10b and confirm that communication has been completed and transition to channel halt mode has been made, and then set the CHMDC[1:0] bits to 01b.

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode. The CAN bus status can be confirmed with the BLF flag of the RSCFDn(CFD)CmERFL register that becomes 1 when dominant lock is detected.

Note 3. In classical CAN mode, when the transition from channel reset mode to channel halt mode is to be made, set the RSCANnCmCFG register in channel reset mode and then shift to channel halt mode. In CANFD mode, set the RSCFDnCFDCmNCFG register and the RSCFDnCFDCmDCFG register, and then make a transition.

28.6.2.4 Channel Communication Mode

In channel communication mode, CAN communication is performed. Each channel has the following communication states during CAN communication.

- Idle: Neither reception nor transmission is in progress.
- Reception: Receiving a message sent from another node.
- Transmission: Transmitting a message.
- Bus off: Isolated from CAN communication.

When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 00b, the channel transitions to channel communication mode. After that, once 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCFDn(CFD)CmSTS register (m = 0, 1) is set to 1 (communication is ready) and transmission and reception are enabled on the CAN network as an active node. At this time, transmission and reception of messages can be started.

28.6.2.5 Bus Off State

A channel transitions to the bus off state according to the transmit/receive error counter increment/decrement rules of the CAN specifications.

The conditions for returning from the bus off state are determined by the BOM[1:0] bits in the RSCFDn(CFD)CmCTR register.

- When BOM[1:0] = 00b:
  Bus off recovery is compliant with the CAN specifications. After 11 consecutive recessive bits have been detected 128 times, a channel returns from the bus off state to the CAN communication ready state (error active state). At that time, the TEC[7:0] and REC[7:0] bits in the RSCFDn(CFD)CmSTS register are initialized to H’00, the BORF flag in the RSCFDn(CFD)CmERFL register is set to 1 (bus off recovery is detected), and a bus off recovery interrupt request is generated. When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 10b (channel halt mode) in the bus off state, the channel transitions to channel halt mode after bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times).
- When BOM[1:0] = 01b:
  When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to 10b and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to H’00. The BORF flag is not set to 1, and bus off recovery interrupt request is not generated.
- When BOM[1:0] = 10b:
  When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to 10b. After bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times), the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to H’00, the BORF flag is set to 1, and a bus off recovery interrupt request is generated.
- When BOM[1:0] = 11b:
  When the CHMDC[1:0] bits are set to 10b in the bus off state, the channel transitions to channel halt mode before bus off recovery is completed. At that time, the TEC[7:0] and REC[7:0] bits are initialized to H’00, but the BORF flag is not set to 1. Also, a bus off recovery interrupt is not generated.
  However, the BORF flag becomes 1 and a bus off recovery interrupt request is generated if a CAN module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to 10b.

If the RS-CANFD module causes the channel to transition to channel halt mode simultaneously with a program write to the CHMDC[1:0] bits, the program write takes precedence. An automatic transition to channel halt mode when the BOM[1:0] bits are set to 01b or 10b is made only when the CHMDC[1:0] bits are 00b (channel communication mode). Furthermore, setting the RTBO bit in the RSCFDn(CFD)CmCTR register to 1 allows a forced return from the bus off state. As soon as the RTBO bit is set to 1, the state changes to the error active state. After 11 consecutive recessive bits have been detected, the CAN module becomes ready for communication. In this case, the BORF flag is not set to 1 and the TEC[7:0] and REC[7:0] bits are initialized to H’00. Write 1 to the RTBO bit only when the BOM[1:0] value is 00b. Writing the RTBO bit to 1 in a state other than the bus off state is ignored, and the RTBO bit is immediately set to 0.

28.6.3 Initializing Registers by Transition to CAN Mode

Table 28.37 lists bits and flags to be initialized by a transition to channel reset mode. These bits and flags are also initialized by a transition to global reset mode. Furthermore, **Table 28.38** lists bits and flags to be initialized only by a transition to global reset mode.

Table 28.37 Registers Initialized in Global Reset Mode or Channel Reset Mode

Register	Bit/Flag
RSCFDn(CFD)CmCTR register	(ROM), CRCT, CTMS[1:0], CTME, CHMDC[1:0]
RSCFDn(CFD)CmSTS register	CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, (ESIF), REC[7:0], TEC[7:0]
RSCFDn(CFD)CmERFL register	CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF
RSCFDnCFDCmFDCTR register	EOCCLR, SOCCLR
RSCFDnCFDCmFDSTS register	SOC[7:0], EOC[7:0], SOCO, EOCO, TDCVF, TDCR[6:0]
RSCFDnCFDCmFDCRC register	CRCREG[20:0], SCNT[3:0]
RSCFDn(CFD)CFCCk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE
RSCFDn(CFD)CFSTSk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF
RSCFDn(CFD)CFTISTS register	CFkTXIF
RSCFDn(CFD)TMCp register	TMOM, TMTAR, TMTR
RSCFDn(CFD)TMSTSp register	TMTARM, TMTRM, TMTRF[1:0], TMTSTS
RSCFDn(CFD)TMTRSTSy register	TMTRSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTARSTSy register	TMTARSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTCSTSy register	TMTCSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTASTSy register	TMTASTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TXQCCm register	TXQE
RSCFDn(CFD)TXQSTSm register	TXQIF, TXQFLL, TXQEMP
RSCFDn(CFD)THLCCm register	THLE
RSCFDn(CFD)THLSTSm register	THLMC[4:0], THLIF, THLELT, THLFLL, THLEMP
RSCFDn(CFD)GTINTSTS0 register	TSIFm, TAIFm, TQIFm, CFTIFm, THIFm (m =0, 1)

Note: Bits and flags in parentheses exist only in registers for use in CANFD mode.

Table 28.38 Registers Initialized Only in Global Reset Mode

Register	Bit/Flag
RSCFDn(CFD)GSTS register	GHLTSTS
RSCFDn(CFD)GERFL register	EEF0, EEF1, (CMPOF), THLES, MES, DEF
RSCFDn(CFD)GTSC register	TS[15:0]
RSCFDn(CFD)RMNDy register	RMNSq
RSCFDn(CFD)RFCCx register	RFE
RSCFDn(CFD)RFSTSx register	RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP
RSCFDn(CFD)CFCCk register	When transmit/receive FIFO buffer is in receive mode: CFE
RSCFDn(CFD)CFSTSk register	When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT
RSCFDn(CFD)FESTS register	CFkEMP, RFxEMP
RSCFDn(CFD)FFSTS register	CFkFLL, RFxFLL
RSCFDn(CFD)FMSTS register	CFkMLT, RFxMLT
RSCFDn(CFD)RFISTS register	RFxIF
RSCFDn(CFD)CFRISTS register	CFkRXIF
RSCFDnCFDCDTCT register	CFDMAEm, RFDMAEx
RSCFDnCFDCDTSTS register	CFDMASTSm, RFDMASTSx
RSCFDn(CFD)GTSTCFG register	RTMPS[6:0], C0ICBCE, C1ICBCE
RSCFDn(CFD)GTSTCTR register	RTME, ICBCTME

Note: Bits and flags in parentheses exist only in registers for use in CANFD mode.

28.7 Reception Function

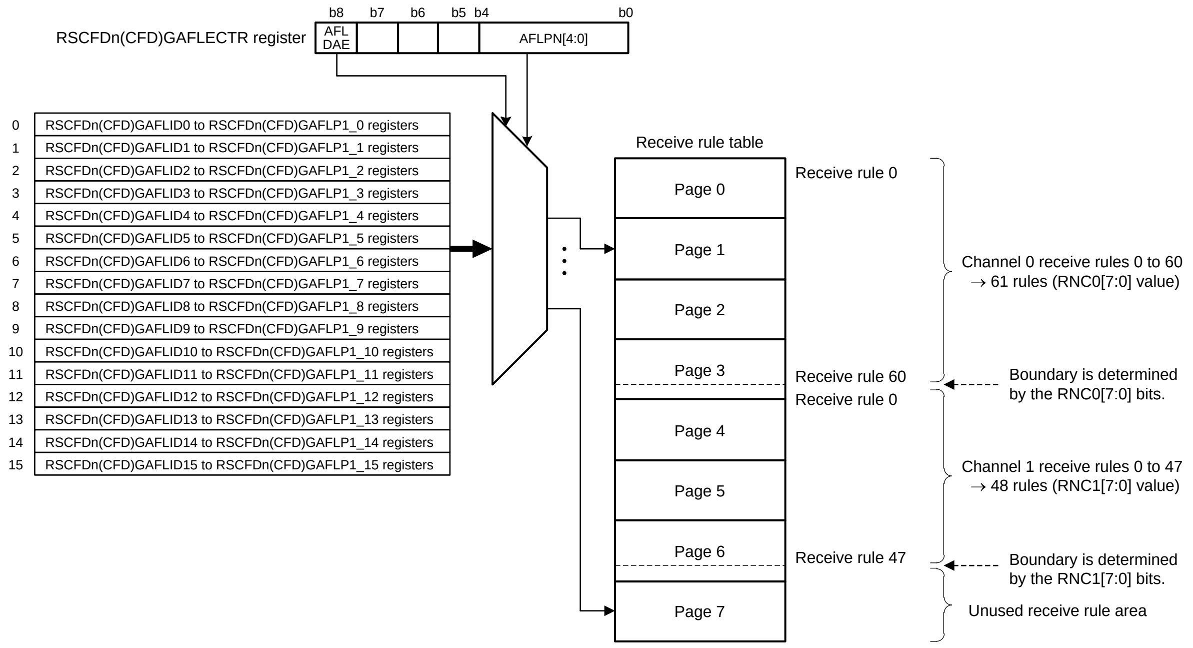
There are two reception types.

- Reception by receive buffers:
  Zero to 32 receive buffers can be shared by all channels. Since messages stored in receive buffers are overwritten at each reception, the latest receive data can always be read.
- Reception by receive FIFO buffers and transmit/receive FIFO buffers (receive mode):
  Eight receive FIFO buffers can be shared by all channels and three dedicated transmit/receive FIFO buffers are provided for each channel. Messages of up to the number of buffer stages specified with the RFDC[2:0] and CFDC[2:0] bits can be stored in FIFO buffers and can be read sequentially from the oldest.

28.7.1 Data Processing Using the Receive Rule Table

Data processing using the receive rule table allows dispatching of selected messages to the specified buffer. Data processing includes acceptance filter processing, DLC filter processing, routing processing, label addition processing, and mirror function processing.

Up to 128 receive rules can be registered per channel and up to (64 × number of channels) total receive rules can be registered in the entire module. (Up to 128 receive rules can be registered in this module that has two channels.) Set receive rules for each channel. Receive rules cannot be shared with other channels. If receive rules are not set, no messages can be received. **Figure 28.6** illustrates how receive rules are registered.



Note: RNC0[7:0], RNC1[7:0]: Bits in the RSCFDn(CFD)GAFLCFG0 register

Figure 28.6 Entry of Receive Rules (for Setting Channels 0 and 1)

NOTE

Receive rules for each channel must be set in contiguous blocks.

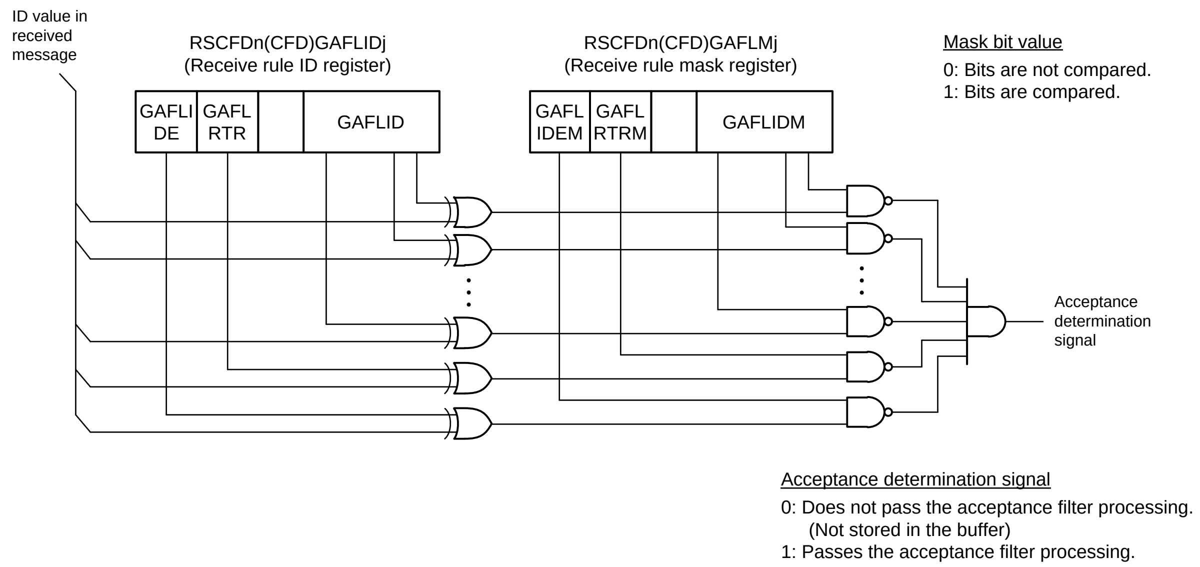
Channel 1 rules and channel 0 rules must be set separately.

Each receive rule consists of 16 bytes in the RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, and RSCFDn(CFD)GAFLP1_j registers (j = 0 to 15). The RSCFDn(CFD)GAFLIDj register is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function, the RSCFDn(CFD)GAFLMj register is used to set mask, the RSCFDn(CFD)GAFLP0_j register is used to set label information to be added, DLC value, and storage receive buffer, and the RSCFDn(CFD)GAFLP1_j register is used to set storage FIFO buffer. Up to 16 receive rules can be set per page.

28.7.1.1 Acceptance Filter Processing

In the acceptance filter processing, the ID data, IDE bit, and RTR bit in a received message are compared with the ID data, IDE bit, and RTR bit set in the receive rule of the corresponding channel. When all these bits match, the message passes through the acceptance filter processing. The ID data, IDE bit, and RTR bit in the received message which correspond to the bits set to 0 (bits are not compared) in the RSCFDn(CFD)GAFLMj register are not compared and are regarded as matched.

Check begins with the receive rule of the minimum number for the corresponding channel. When all the bits to be compared in a received message match the bits set in the receive rule or when all the receive rules are compared without any match, filter processing stops. If there is no matching receive rule, the received message is not stored in the receive buffer or FIFO buffer.



Mask bit value
0: Bits are not compared.
1: Bits are compared.

Acceptance determination signal
0: Does not pass the acceptance filter processing.
(Not stored in the buffer)
1: Passes the acceptance filter processing.

Note: j = 0 to 15
GAFLIDE, GAFLRTR, GAFLID: Bits in the RSCFDn(CFD)GAFLIDj register
GAFLIDEM, GAFLRTRM, GAFLIDM: Bits (mask bits) in the RSCFDn(CFD)GAFLMj register

Figure 28.7 Acceptance Filter Function

28.7.1.2 DLC Filter Processing

When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), DLC filter processing is added to messages that passed through the acceptance filter processing. When the DLC value in a message is equal to or larger than the DLC value set in the receive rule, the message passes through the DLC filter processing.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 0 (DLC replacement is disabled), the DLC value in the received message is stored in the buffer. In this case, all the data bytes in the received message are stored in the buffer.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 1 (DLC replacement is enabled), the DLC value in the receive rule is stored in the buffer instead of the DLC value in the received message. In this case, a value of H’00 is stored in each data byte beyond the number of bytes which is indicated by the DLC value in the receive rule.

When the DLC value in the received message is smaller than that in the receive rule, the message does not pass through the DLC filter processing. In this case, the message is not stored in the receive buffer or the FIFO buffer and the DEF flag in the RSCFDn(CFD)GERFL register is set to 1 (a DLC error is present).

28.7.1.3 Routing Processing

Messages that passed through the acceptance filter processing and the DLC filter processing are stored in receive buffers, receive FIFO buffers, or transmit/receive FIFO buffers (set to receive mode or gateway mode). Message storage destination is set by the GAFLRMV and GAFLRMDP[6:0] bits in the RSCFDn(CFD)GAFLP0_j register (j = 0 to 15) and by the RSCFDn(CFD)GAFLP1_j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to eight buffers.

In CANFD mode, if the payload length of the received message exceeds the payload storage size of the storage buffer, the CMPOF flag in the RSCFDnCFDGERFL register is set to 1 (payload overflow) and the processing is handled according to the CMPOC bit in the RSCFDnCFDGCFG register. When the CMPOC bit is 0, the received message which exceeds the payload storage size is not stored in the buffer. When the CMPOC bit is 1, the received message is stored in the buffer with payloads exceeding the storage size being discarded.

28.7.1.4 Label Addition Processing

It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits in the RSCFDn(CFD)GAFLP0_j register.

28.7.1.5 Mirror Function Processing

The mirror function allows the CAN node to receive its own transmitted messages. The mirror function is made available by setting the MME bit in the RSCFDn(CFD)GCFG register to 1 (mirror function is enabled).

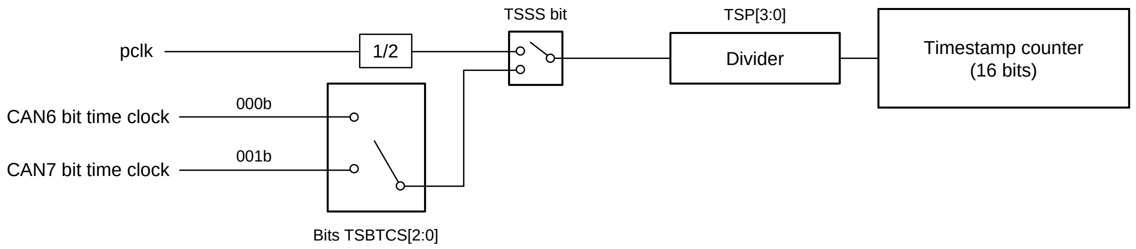
When the mirror function is in use, receive rules for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register is set to 0 are used for data processing when receiving messages transmitted from other CAN nodes. When the CAN node is receiving its own transmitted messages, receive rules for which the GAFLLB bit is set to 1 are used for data processing.

28.7.1.6 Timestamp

The timestamp counter is a 16-bit free-running counter used for recording message receive time and transmission time. The timestamp counter value is fetched at the timing set with the TSCCFG[1:0] bits in the RSCFDn(CFD)GCFG register and is then stored in a receive buffer or a FIFO buffer together with the message ID and data during data reception. The clock source of the timestamp counter is selected by the TSBTCS[2:0] and TSSS bits in the RSCFDn(CFD)GCFG register. In classical CAN mode, either pclk/2 or the CANm bit time clock (m = 0, 1). In CANFD mode, the clock source is selectable from pclk/2 or nominal CANm bit time clock. However, do not select the nominal CANm bit time clock of channels that handle CANFD frames. The timestamp counter count source is obtained by dividing the selected clock source by the TSP[3:0] value in the RSCFDn(CFD)GCFG register.

When the CANm bit time clock or nominal CANm bit time clock is used as a clock source, the timestamp counter stops when the corresponding channel transitions to channel reset mode or channel halt mode. When the pclk/2 is used as a clock source, the timestamp function is not affected by channel mode.

The timestamp counter value is reset to H’0000 by setting the TSRST bit in the RSCFDn(CFD)GCTR register to 1.



Note: TSBTCS[2:0], TSSS, TSP[3:0]: Bits in the RSCFDn(CFD)GCFG register
When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000b.

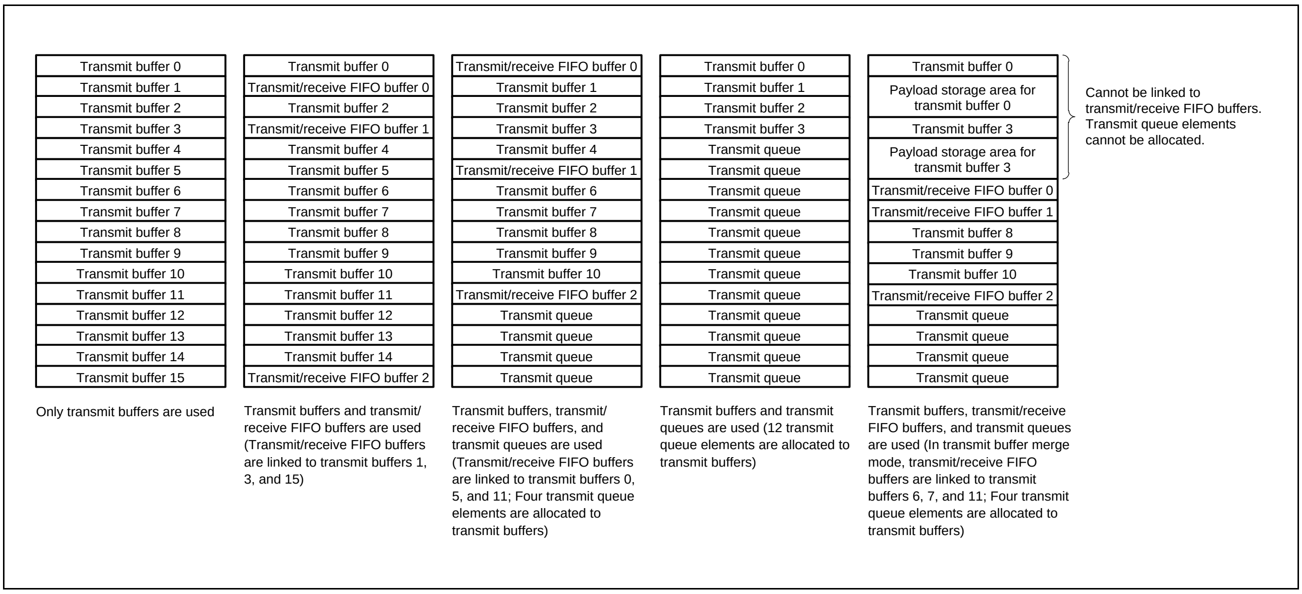
Figure 28.8 Timestamp Function Block Diagram

28.8 Transmission Functions

There are three types of transmission. In classical CAN mode, transmittable payload length is 8 bytes in every transmission type. In CANFD mode, transmittable payload length varies with transmission types.

- Transmission using transmit buffers:
  Each channel has 16 buffers. Transmittable payload length in CANFD mode is 20 bytes. However, when transmit buffer merge mode is used, four buffers out of 16 buffers are allocated as a payload-only storage area and two buffers are able to transmit payloads with a length of more than 20 bytes.
- Transmission using transmit/receive FIFO buffers (transmit mode):
  Each channel has three FIFO buffers. Up to 128 messages can be contained in a single FIFO buffer. Transmittable payload length in CANFD mode is 64 bytes. Each FIFO buffer is used with a link to a transmit buffer. Only the message to be transmitted next in a FIFO buffer becomes the target of transmit priority determination. Messages are transmitted sequentially on a first-in, first-out basis.
- Transmission using transmit queues:
  Up to 16 transmit buffers per channel can be allocated to the transmit queues. Transmittable payload length in CANFD mode is 20 bytes. Transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel. Transmit buffers are allocated to transmit queues in descending order of buffer number. All messages in transmit queues, which are targets of priority determination, are transmitted in the order of ID number.

Figure 28.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.



Transmit buffer 0	Transmit buffer 0	Transmit/receive FIFO buffer 0	Transmit buffer 0	Transmit buffer 0
Transmit buffer 1	Transmit/receive FIFO buffer 0	Transmit buffer 1	Transmit buffer 1	Payload storage area for transmit buffer 0
Transmit buffer 2	Transmit buffer 2	Transmit buffer 2	Transmit buffer 2	
Transmit buffer 3	Transmit/receive FIFO buffer 1	Transmit buffer 3	Transmit buffer 3	Transmit buffer 3
Transmit buffer 4	Transmit buffer 4	Transmit buffer 4	Transmit queue	Payload storage area for transmit buffer 3
Transmit buffer 5	Transmit buffer 5	Transmit/receive FIFO buffer 1	Transmit queue	
Transmit buffer 6	Transmit buffer 6	Transmit buffer 6	Transmit queue	Transmit/receive FIFO buffer 0
Transmit buffer 7	Transmit buffer 7	Transmit buffer 7	Transmit queue	Transmit/receive FIFO buffer 1
Transmit buffer 8	Transmit buffer 8	Transmit buffer 8	Transmit queue	Transmit buffer 8
Transmit buffer 9	Transmit buffer 9	Transmit buffer 9	Transmit queue	Transmit buffer 9
Transmit buffer 10	Transmit buffer 10	Transmit buffer 10	Transmit queue	Transmit buffer 10
Transmit buffer 11	Transmit buffer 11	Transmit/receive FIFO buffer 2	Transmit queue	Transmit/receive FIFO buffer 2
Transmit buffer 12	Transmit buffer 12	Transmit queue	Transmit queue	Transmit queue
Transmit buffer 13	Transmit buffer 13	Transmit queue	Transmit queue	Transmit queue
Transmit buffer 14	Transmit buffer 14	Transmit queue	Transmit queue	Transmit queue
Transmit buffer 15	Transmit/receive FIFO buffer 2	Transmit queue	Transmit queue	Transmit queue
Only transmit buffers are used	Transmit buffers and transmit/receive FIFO buffers are used (Transmit/receive FIFO buffers are linked to transmit buffers 1, 3, and 15)	Transmit buffers, transmit/receive FIFO buffers, and transmit queues are used (Transmit/receive FIFO buffers are linked to transmit buffers 0, 5, and 11; Four transmit queue elements are allocated to transmit buffers)	Transmit buffers and transmit queues are used (12 transmit queue elements are allocated to transmit buffers)	Transmit buffers, transmit/receive FIFO buffers, and transmit queues are used (In transmit buffer merge mode, transmit/receive FIFO buffers are linked to transmit buffers 6, 7, and 11; Four transmit queue elements are allocated to transmit buffers)

Cannot be linked to transmit/receive FIFO buffers. Transmit queue elements cannot be allocated.

Figure 28.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

28.8.1 Transmit Priority Determination

If transmit requests are issued from multiple buffers or from the queue on the same channel, transmit priority is determined using one of the following methods.

The priority is determined by using one of the following methods.

- ID priority (TPRI bit = 0)
- Transmit buffer number priority (TPRI bit = 1)

All CAN channels use the setting of the TPRI bit in the RSCFDn(CFD)GCFG register.

When the TPRI bit is set to 0, messages are transmitted according to the priority of stored message IDs. ID priority conforms to the CAN bus arbitration specification defined in the CAN specifications. All IDs of pending transmit messages are targets of priority determination, regardless of whether they are stored in transmit buffers, transmit/receive FIFO buffers (set to transmit mode or gateway mode), or the transmit queue. If even a single transmit queue is used, select ID priority. When transmit/receive FIFO buffers are used, the oldest message in a FIFO buffer becomes the target of priority determination. When a message is being transmitted from a transmit/receive FIFO buffer, the next message in the FIFO buffer becomes the target of priority determination. When a transmit queue is used, all messages in the transmit queue are targets of priority determination. If the same ID is set for two or more buffers, the buffer with the smaller buffer number takes precedence.

When the TPRI bit is set to 1, the message in the transmit buffer with the minimum buffer number among all buffers with a transmit request is transmitted first. When transmit/receive FIFO buffers are linked to transmit buffers, transmit priority is determined according to linked transmit buffer numbers.

When messages are retransmitted due to an arbitration-lost or an error, transmit priority determination is made again regardless of the TPRI bit.

28.8.2 Transmission Using Transmit Buffers

Setting the transmit request bit (TMTR bit in the RSCFDn(CFD)TMCp register) in a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.

The transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register (p = 0 to 31). When transmit completes successfully, the TMTRF[1:0] flag is set to 10b (transmission has been completed (without transmit abort request)) or 11b (transmission has been completed (with transmit abort request)).

28.8.2.1 Transmit Abort Function

With respect to transmit buffers for which the TMTRM bit in the RSCFDn(CFD)TMSTSp register is set to 1 (a transmit request is present), when the TMTAR bit in the RSCFDn(CFD)TMCp register is set to 1 (transmit abort is requested), the transmit request is canceled. When transmit abort is completed, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSp register is set to 01b (transmit abort has been completed) and the transmit request is canceled (clearing the TMTRM bit to 0).

A message that is being transmitted or a message to be transmitted next according to the transmit priority determination cannot be aborted. However, when an arbitration-lost or an error occurs during transmission of a message for which the TMTAR bit is set to 1, retransmission is not performed.

28.8.2.2 One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit in the RSCFDn(CFD)TMCp register is set to 1 (one-shot transmission is enabled), transmission is performed only once. Even if an arbitration-lost or an error occurs, retransmission is not performed.

The one-shot transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to 10b or 11b. When an arbitration-lost or an error occurs, the TMTRF[1:0] flag is set to 01b (transmit abort has been completed).

28.8.2.3 Transmit Buffer Merge Mode (Only in CANFD Mode)

Transmit buffers can transmit messages with a payload length of 20 bytes, but can transmit messages with a payload length of up to 64 bytes by merging three transmit buffers in transmit buffer merge mode.

Setting the TMME bit to 1 in the RSCFDnCFDCmFDCFG register enables transmit buffer merge mode. In this mode, six buffers per channel become a merge area and two sets of transmit buffers (16 × m) + 0 to (16 × m) + 2 and transmit buffers (16 × m) + 3 to (16 × m) + 5 are merged. A transmission request is made by the first transmit buffer, and subsequent two buffers are used as a payload storage area. Do not set the transmission request bit (TMTR bit in the RSCFDnCFDTMCp register) and the transmission abort request bit (TMTAR bit in the RSCFDnCFDTMCp register) to 1 for transmit buffers except for the first buffer.

While transmit buffer merge mode is enabled, do not link the transmit/receive FIFO buffer to the merged buffers or allocate it to the transmit queue.

28.8.3 Transmission Using FIFO Buffers

Multiple messages can be stored in a single transmit/receive FIFO buffers, up to the number specified by the FIFO buffer depth, which is set by the CFDC[2:0] bits in the RSCFDn(CFD)CFCCk register (k = 0 to 5). Messages are transmitted sequentially on a first-in, first-out basis.

Each transmit/receive FIFO buffer is linked to a transmit buffer selected by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register. When the CFE bit in the RSCFDn(CFD)CFCCk register is set to 1 (transmit/receive FIFO buffers are used), transmit/receive FIFO buffers become targets of transmit priority determination. Priority of only the next transmit message is determined in the FIFO buffer.

When the CFE bit is set to 0 (no transmit/receive FIFO buffer is used), the CFEMP flag is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately if the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, arbitration-lost or the transition to channel halt mode in the case that a message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages cannot be stored in FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

28.8.3.1 Interval Transmission Function

A message transmission interval time can be set to space the transmission of messages from the same FIFO buffer when using a transmit/receive FIFO buffer set to transmit mode or gateway mode.

Immediately after the first message has been transmitted successfully from the FIFO buffer with the CFE bit in the RSCFDn(CFD)CFCCk register set to 1, the interval timer starts counting (after EOF7 of the CAN protocol). After that, when the interval time has passed, the next message is transmitted. The interval timer stops in channel reset mode or by clearing the CFE bit to 0.

The interval time is set by the CFITT[7:0] bits in the RSCFDn(CFD)CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to H’00.

Select an interval timer count source using the CFITR and CFITSS bits in the RSCFDn(CFD)CFCCk register. When the CFITR and CFITSS bits are set to 00b, the count source is obtained by dividing pclk/2 by the value of the ITRCP[15:0] bits. When the CFITR and CFITSS bits are set to 10b, the count source is obtained by dividing pclk/2 by (the value of the ITRCP[15:0] bits in the RSCFDn(CFD)GCFG register × 10). When the CFITR and CFITSS bits are set to x1b, the CANm bit time clock becomes a count source in classical CAN mode and the nominal CANm bit time clock becomes a count source in CANFD mode. Use this count source only for the channel that does not handle the CANFD frames. The interval time is calculated by the following equations where M is the value of ITRCP[15:0] and N is the value of CFITT[7:0].

- When CFITR and CFITSS = 00b:

$$\frac{1}{\text{pclk frequency}} \times 2 \times \mathrm{M} \times \mathrm{N}$$

- When CFITR and CFITSS = 10b:

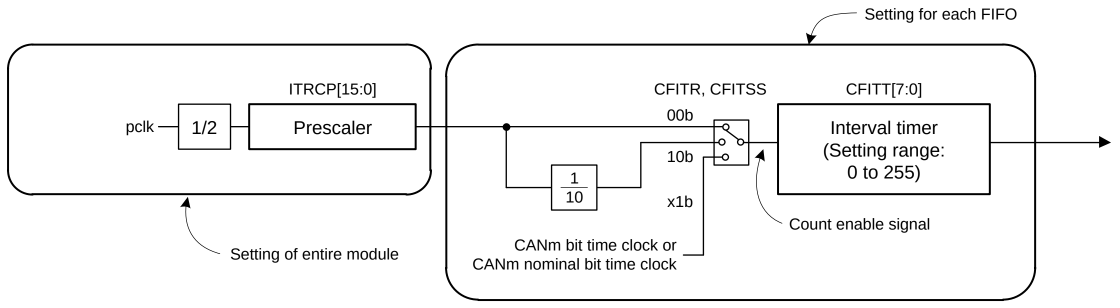
$$\frac{1}{\text{pclk frequency}} \times 2 \times \mathrm{M} \times 10 \times \mathrm{N}$$

- When CFITR and CFITSS = x1b:

Classical CAN mode: $\frac{1}{\text{CANm bit time clock frequency}} \times \text{N}$

CANFD mode: $\frac{1}{\text{Nominal CANm bit time clock frequency}} \times \text{N}$

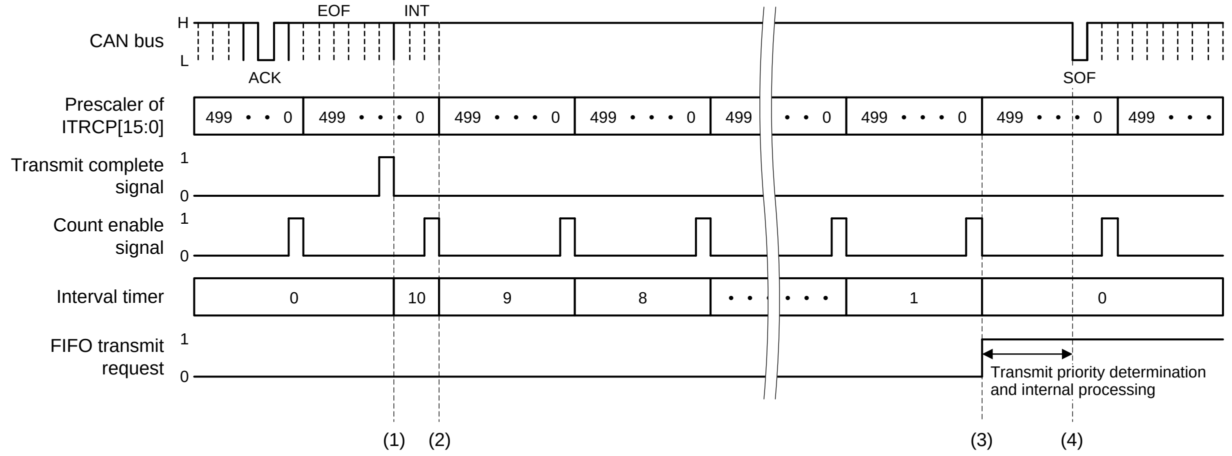
Figure 28.10 shows the interval timer block diagram.



Note: ITRCP[15:0]: Bits in the RSCFDn(CFD)GCFG register
CFITR, CFITSS, CFITT[7:0]: Bits in the RSCFDn(CFD)CFCCk register
m = 0, 1, k = 0 to 5

Figure 28.10 Interval Timer Block Diagram

Figure 28.11 shows the interval timer timing diagram.



$$\text{Interval time (logical value)} = \frac{2}{\text{pclk frequency}} \times \text{(set ITRCP[15:0] value)} \times \text{set CFITT[7:0] value}$$

Note: ITRCP[15:0]: Bits in the RSCFDn(CFD)GCFG register (The set value is 500 in this figure.)
CFITT[7:0]: Bits in the RSCFDn(CFD)CFCCk register (The set value is 10 in this figure.)

Figure 28.11 Interval Timer Timing Chart

(1) The interval timer starts counting upon completion of transmission. Since the prescaler is not initialized at the time of transmission completion, the first interval time contains an error of up to one count of the interval timer.

(2) The interval timer is decremented by the next count enable signal.

(3) When the interval timer has decreased to 0, the transmit/receive FIFO buffer issues a transmit request.

(4) The transmit/receive FIFO buffer is determined for the next transmission by the priority determination, it starts transmitting data. Transmission starts usually with a delay of three CANm bit time clock cycles or less from the issue of transmit request. If multiple internal processes (such as receive filter processing, message routing, and transmit priority determination) take place in all channels, a delay of up to 644 cycles of the pclk may be generated.

28.8.4 Transmission Using Transmit Queues

Three to sixteen buffers (up to 10 buffers in transmit buffer merge mode) are allocated to a transmit queue for each channel, and transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel.

All messages in a transmit queue are targets of transmit priority determination and are transmitted in the ID priority order regardless of storage sequence. If two messages having the same ID are stored in a transmit queue, these messages are not always transmitted in the order of their storage in the transmit queue.

Setting the TXQE bit in the RSCFDn(CFD)TXQCCm register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCFDn(CFD)TXQSTSm register is set to 1 (the transmit queue contains no messages (transmit queue empty)) at the timing below.

- The transmit queue becomes empty immediately when no message in it is being transmitted or will be transmitted next.
- The transmit queue becomes empty after transmission completion, CAN bus error detection, arbitration-lost, or the transition to channel halt mode when a message in it is being transmitted or will be transmitted next.

When the TXQE bit is cleared to 0, all messages in transmit queues are lost and messages cannot be stored in transmit queues. Confirm that the TXQEMP flag is set to 1 before setting the TXQE bit to 1 again.

28.8.5 Transmit Data Padding (Only in CANFD Mode)

When the payload length indicated by the set DLC value in a transmit message exceeds the payload storage area size of a buffer to be used for transmission, excessive payloads are padded by H’CC.

This processing is performed in the following cases when the transmit buffer merge mode is disabled (TMME bit in the RSCFDnCFDCmFDCFG register is 0).

- Transmit/receive FIFO set to transmission or gateway mode:
  When the payload length of the transmit DLC exceeds the transmit/receive FIFO payload storage area size set by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register
- Transmit buffer (including transmit queue):
  When the payload length of the transmit DLC exceeds 20 bytes

When the transmit buffer merge mode is enabled, no transmit data is padded in any transmission using a transmit buffer, transmit/receive FIFO buffer, or transmit queue. At this time, do not set a payload length more than the payload storage size of the buffer for transmitting as the DLC value in the transmit message.

28.8.6 Transmit History Function

Information about transmission-completed messages can be stored in the transmit history buffer. Each channel has a single transmit history buffer that can contain 16 sets of transmit history data.

A message transmit source buffer type can be selected by the THLDTE bit in the RSCFDn(CFD)THLCCm register. The THLEN bit in the RSCFDn(CFD)CFIDk register (k = 0 to 5) determines whether transmit history data is stored for each message.

In classical CAN mode, the TMTSCE bit in the RSCANnGCFG register can be used to set whether to include a timestamp value in the transmit history data. In CANFD mode, a timestamp value is always included.

The following information on a transmitted message will be stored in the transmit history buffer after the successful completion of transmission.

Storage of the transmit history data after the successful completion of transmission may take up to 128 cycles of pclk in classical CAN mode or 396 cycles of pclk in CANFD mode.

- Buffer type: 001b: Transmit buffer
  010b: Transmit/receive FIFO buffer
  100b: Transmit queue
- Buffer number: Number of source transmit buffer, transmit queue, or transmit/receive FIFO buffer.
  This number depends on buffer types. See **Table 28.39**.
- Label data: Label information of the transmit message
- Timestamp: Timestamp value of the transmit message
  (When the TMTSCE bit is 1 in classical CAN mode)

Table 28.39 Transmit History Data Buffer Numbers

Buffer Type Buffer No.	001b	010b	100b
0000b	Transmit buffer 16 × m + 0	Buffer numbers of the transmit buffer linked to the transmit/receive FIFO buffer by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register (k = 0 to 5)	Buffer numbers of the transmit buffer allocated to the transmit queue that performed transmission
0001b	Transmit buffer 16 × m + 1		
0010b	Transmit buffer 16 × m + 2		
0011b	Transmit buffer 16 × m + 3		
0100b	Transmit buffer 16 × m + 4		
0101b	Transmit buffer 16 × m + 5		
0110b	Transmit buffer 16 × m + 6		
0111b	Transmit buffer 16 × m + 7		
1000b	Transmit buffer 16 × m + 8		
1001b	Transmit buffer 16 × m + 9		
1010b	Transmit buffer 16 × m + 10		
1011b	Transmit buffer 16 × m + 11		
1100b	Transmit buffer 16 × m + 12		
1101b	Transmit buffer 16 × m + 13		
1110b	Transmit buffer 16 × m + 14		
1111b	Transmit buffer 16 × m + 15		

Label data is used to identify each message. Unique label data can be added to each message transmitted from a transmit buffer, transmit queue, or transmit/receive FIFO buffer.

The timestamp value is fetched from the timestamp counter at the SOF (start of frame) timing of the message. For details about the timestamp counter, see **Section 28.7.1.6, Timestamp**.

Transmit history data can be read from the RSCFDn(CFD)THLACCm register. If an attempt is made to store new transmit history data while the buffer is full, the buffer overflows and the new data is discarded.

28.9 Gateway Function

When a transmit/receive FIFO buffer is set to gateway mode, receive messages can be transmitted from an arbitrary channel without CPU intervention.

When the CFM[1:0] bits in the RSCFDn(CFD)CFCCk register are set to 10b (gateway mode) for the transmit/receive FIFO buffer selected by the RSCFDn(CFD)GAFLP1_j register of a channel being used for transmission, messages that pass through filter processing according to the receive rule are stored in the specified transmit/receive FIFO buffer and are automatically transmitted from the buffer.

Messages stored in a transmit/receive FIFO buffer are transmitted sequentially on a first-in, first-out basis. Only the message to be transmitted next becomes the target of transmit priority determination.

Transmit/receive FIFO buffers in the gateway mode are disabled by setting the CFE bit in the RSCFDn(CFD)CFCCk register to 0 and the CFEMP flag becomes 1 according to the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the message in it is not being transmitted and will not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when the message in it is being transmitted or will be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages can no longer be stored in transmit/receive FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

28.9.1 CAN-CANFD Gateway (Only in CANFD Mode)

When the gateway function is used in CANFD mode, a frame to be transmitted can be replaced with a classical CAN frame or a CANFD frame.

Setting the GWEN bit in the RSCFDnCFDCmFDCFG register to 1 enables the CAN-CANFD gateway. The FDF and BRS bits in the transmit frame can be selected by the GWFDF and GWBRS bits in the RSCFDnCFDCmFDCFG register. When the DLC value of the received CAN frame is 1001b or more and the GWFDF bit is 1 (CANFD frame), the DLC value is replaced with 1000b.

When the CAN-CANFD gateway is enabled, do not perform routing for the following frames.

- CANFD frames with a payload length of more than 8 bytes
- Remote frames

When the CAN-CANFD gateway is enabled, the following frame should be transmitted in the channel by setting of GWFDF.

- When GWFDF bit is set to 0, only classical CAN frames should be transmitted.
- When GWFDF bit is set to 1, only CANFD frames should be transmitted.

28.10 Test Function

The test function is classified into communication tests and global tests.

- Communication tests: Performed for each channel.
  - Standard test mode
  - Listen-only mode
  - Self-test mode 0 (external loopback mode)
  - Self-test mode 1 (internal loopback mode)
  - Restricted operation mode (only in CANFD mode)
- Global tests: Performed for the entire module
  - RAM test (read/write test)
  - Inter-channel communication test [CRC error test enabled]

28.10.1 Standard Test Mode

CRC tests are enabled in standard test mode. The CRC value calculated by the RS-CANFD module based on the transmit message or receive message is stored in the register. This CRC value is stored in the CRCREG[14:0] bits in the RSCFDn(CFD)CmERFL register when the message is a classical CAN frame (CRC length = 15 bits) or in the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register when the message is a CANFD frame (CRC length = 17 or 21 bits). Use the inter-channel communication test function for CRC error tests. For details, see **Section 28.10.6.1, CRC Error Test**.

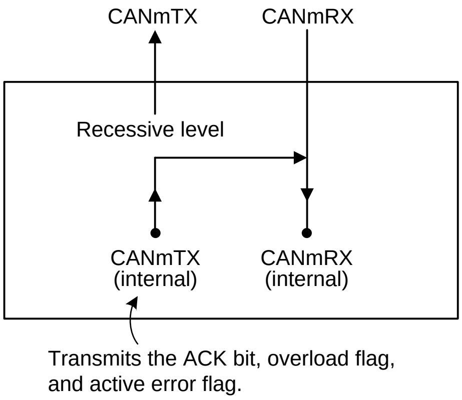
28.10.2 Listen-Only Mode

Listen-only mode allows reception of data frames and remote frames. Only recessive bits are transmitted on the CAN bus, and the ACK bit, overload flag, and active error flag are not transmitted.

Listen-only mode is available for detecting the communication speed.

Do not make a transmit request from any buffer or queue in listen-only mode.

Figure 28.12 shows the connection when listen-only mode is selected.



Note: m = 0, 1

Figure 28.12 Connection when Listen-Only Mode is Selected

28.10.3 Self-Test Mode (Loopback Mode)

In self-test mode, transmitted messages are compared with the receive rule of the own channel and the messages are stored in a buffer if they have passed through the filter processing. Messages transmitted from other CAN nodes are compared only with the receive rule for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register (j = 0 to 15) is set to 0 (when a message transmitted from another CAN node is received).

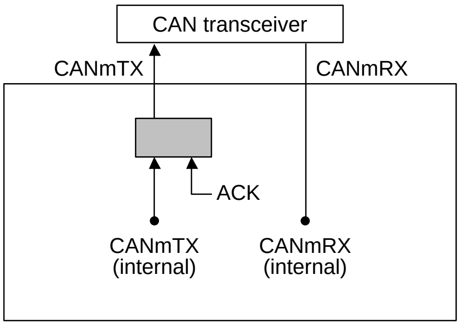
If the mirror function and self-test mode are both enabled, the self-test mode setting takes precedence.

28.10.3.1 Self-Test Mode 0 (External Loopback Mode)

Self-test mode 0 is used to perform a loopback test within a channel including the CAN transceiver.

In self-test mode 0, transmitted messages are handled as messages received through the CAN transceiver and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

Figure 28.13 shows the connection when self-test mode 0 is selected.



Note: m = 0, 1

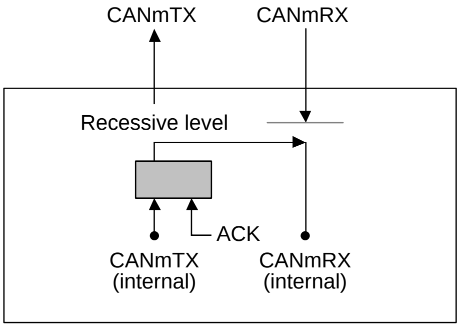
Figure 28.13 Connection when Self-Test Mode 0 is Selected

28.10.3.2 Self-Test Mode 1 (Internal Loopback Mode)

In self-test mode 1, transmitted messages are handled as received messages and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

In self-test mode 1, internal feedback from the internal CANmTX pin (m = 0, 1) to the internal CANmRX pin is performed. The external CANmRX pin input is isolated. The external CANmTX pin outputs only recessive bits.

Figure 28.14 shows the connection when self-test mode 1 is selected.



Note: m = 0, 1

Figure 28.14 Connection when Self-Test Mode 1 is Selected

28.10.4 Restricted Operation Mode (Only in CANFD Mode)

In restricted operation mode, an ACK bit is generated when a valid data frame and a remote frame have been received, but these frames are not transmitted even if an error frame or an overload frame transmit condition is detected. When a condition is detected, operation is suspended until the bus idle state comes for resynchronization with the CAN communication. The receive error counter (REC) and the transmit error counter (TEC) do not change due to an error.

A desired transmission request can be made for transmission without restrictions.

28.10.5 RAM Test

The RAM test function allows accesses to all CAN RAM addresses.

When the RAM test function is used, the RAM is divided into pages of 256 bytes each. RAM test page is set by the RTMPS[6:0] bits in the RSCFDn(CFD)GTSTCFG register. Data in the set page can be read from and written to the RSCFDn(CFD)RPGACCr register (r = 0 to 63). The available total RAM size is 5120 bytes (H’1400) in classical CAN mode or 7104 bytes (H’1BC0) in CANFD mode. Do not access the RAM area higher than the 192th byte on the last page (the setting of the RTMPS bit is H’1B) in CANFD mode.

28.10.6 Inter-Channel Communication Test

The inter-channel communication test function allows communication test by internally connecting CAN channels to each other. During this test, channels are isolated from the external CAN bus.

Before starting data transmission/reception in channel communication mode, make transmission/reception settings for each channel.

Figure 28.15 shows the connection for inter-channel communication test.

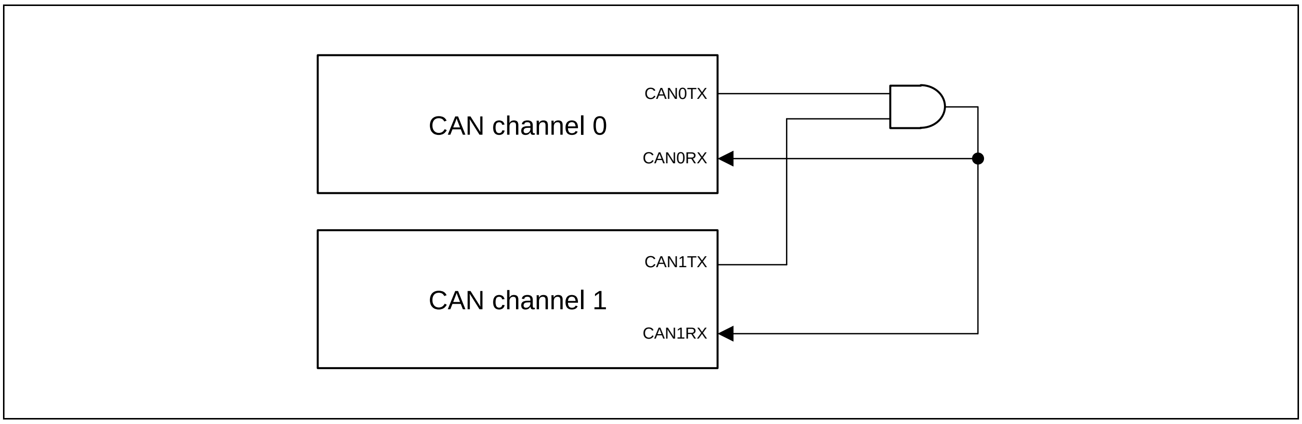


Figure 28.15 Connection for Inter-Channel Communication Test

28.10.6.1 CRC Error Test

A CRC error test is enabled during an inter-channel communication test. The following shows an example of channel 0 CRC error test procedure during a communication test between channel 0 and channel 1.

Preconditions

- Inter-channel communication test is enabled.
- Channel 0 and channel 1 are in standard test mode.

Procedure

1. Make a setting to send a message from the transmit buffer p of channel 1.
2. Set the CRCT bit in the RSCFDn(CFD)C0CTR register to 1 (to enable inversion of the first bit in the received ID field).
3. Set the TMTR bit in the RSCFDn(CFD)TMCp register to 1 (to issue a transmission request to the transmit buffer p of channel 1).
4. Wait for occurrence of a CAN0 error interrupt due to a channel bus error.
5. Read the CRCREG[14:0] bits in the RSCFDn(CFD)CmERFL register or the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register of channel 0 and channel 1, and confirm that the CRC values are different on the transmission and the reception side.
6. Confirm that the CERR bit in RSCFDm(CFD)C0ERFL is 1 (CRC error detected).

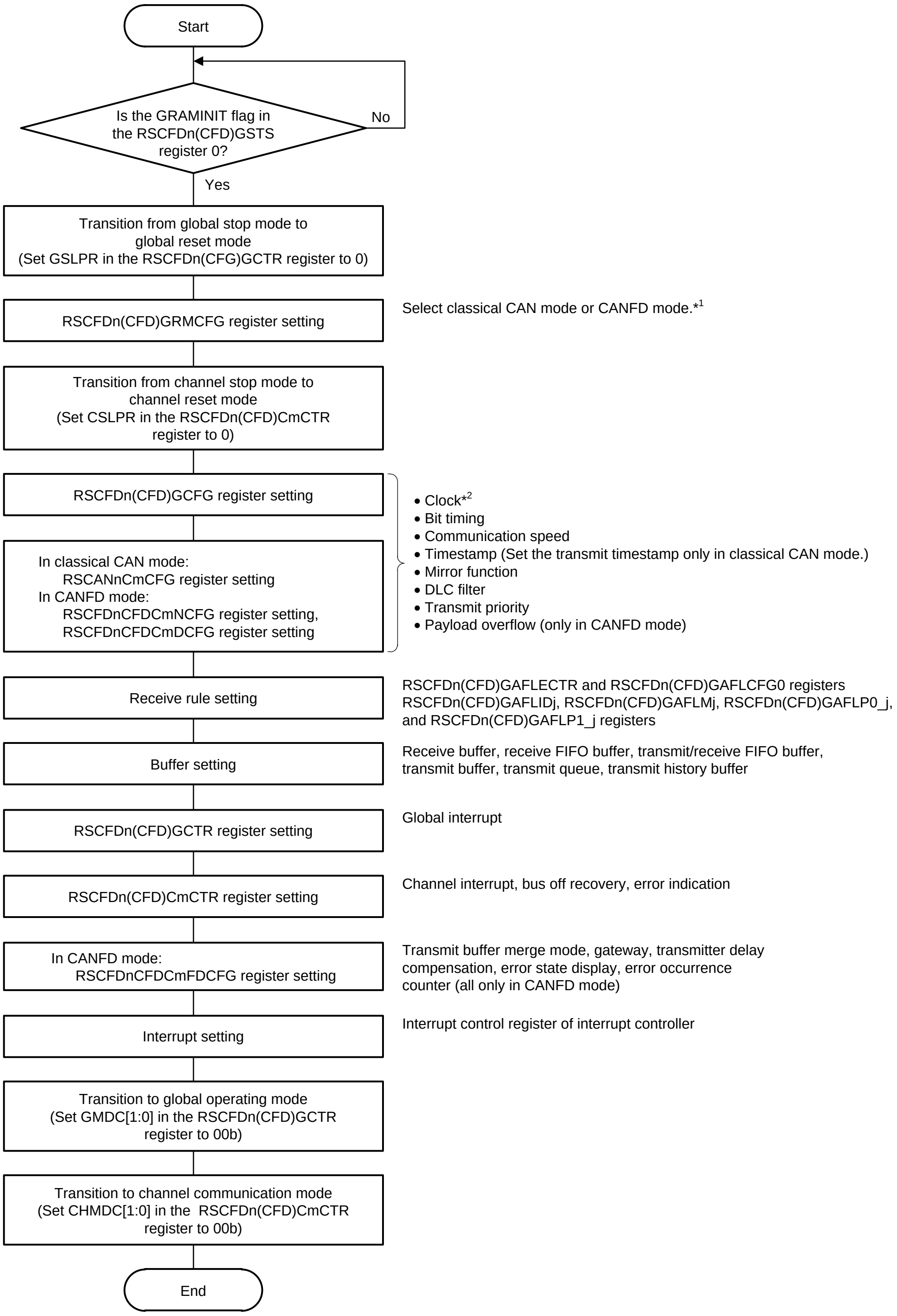
The CRC error test function generates an incorrect CRC value by inverting the first bit in the received ID field. Therefore, note that not a CRC error but a stuff error (continuous 6-bit data of the same level) is detected when a message in which ID’s upper 5-bit value is 10000b or ID’s upper 6-bit value is 011111b is received.

The CRC generation circuit of the RS-CANFD module is contained in the protocol controller of each channel. Another CRC calculation test is not necessary during transmission because the same circuit is used for both transmission and reception.

28.11 RS-CANFD Setting Procedure

28.11.1 Initial Settings

The RS-CANFD module initializes the CAN RAM after the MCU is reset. The RAM initialization time is 2530 cycles of the pclk. The GRAMINIT flag in the RSCFDn(CFD)GSTS register is set to 1 (CAN RAM initialization is ongoing) during the RAM initialization and is cleared to 0 (CAN RAM initialization is finished) when the initialization is completed. Make CAN settings after the GRAMINIT flag is cleared to 0. **Figure 28.16** shows the CAN setting procedure after the MCU is reset.



Note: m = 0, 1, j = 0 to 15

Note 1. The RSCANnGRMCFG register and the RSCFDnCFDGRMCFG register are the same register. Therefore, set either one of the registers.

Note 2. For the setting of CAN clock frequency, see **Section 28.1.3, Clock Supply**.

Figure 28.16 CAN Setting Procedure after the MCU is Reset

28.11.1.1 Clock Setting

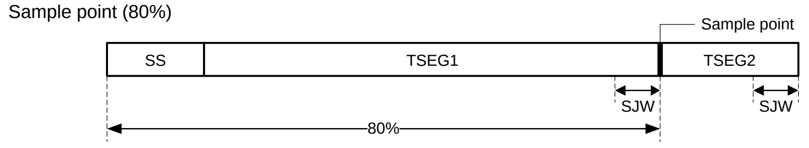
Set the CAN clock (fCAN) as a clock source of the RS-CANFD module. Select the clk_xincan or clkc using the DCS bit in the RSCFDn(CFD)GCFG register.

28.11.1.2 Bit Timing Setting

In the CAN protocol, one bit of a communication frame consists of three segments SS, TSEG1, and TSEG2, of which two segments TSEG1 and TSEG2 can be set by the corresponding registers for each channel. In classical CAN mode, set these two segments in the RSCANnCmCFG register. Two bit rates (nominal bit rate and data bit rate) are provided for CANFD mode. Set the nominal bit rate in the RSCFDnCFDCmNCFG register and set the data bit rate in the RSCFDnCFDCmDCFG register. Sample point timing can be determined by setting these two segments. This timing can be adjusted in units of 1 Time Quantum (hereafter referred to as Tq). A single Tq is the cycle of clock obtained by dividing the clock selected by the DCS bit in the RSCFDn(CFD)GCFG register. Set a division ratio by the BRP[9:0] bits in the RSCANnCmCFG register in classical CAN mode (CANmTq clock), and by the NBRP[9:0] bits in the RSCFDnCFDCmNCFG register and the DBRP[7:0] bits in the RSCFDnCFDCmDCFG register in CANFD mode (CANmTq(N) clock and CANmTq(D) clock). Be sure to specify the same values for both NBRP[9:0] and DBRP[7:0]. To specify different values for the nominal bit rate and the data bit rate, change the values of the RSCFDnCFDCmNCFG.NTSEG1 and NTSEG2 bits and RSCFDnCFDCmDCFG.DTSEG1 and DTSEG2 bits, respectively.

When the TDCE bit is set to 1 (Transmitter delay compensation is enabled) in the RSCFDnCFDCmFDCFG register, set the equal value of 1 or less to the bits NBRP[9:0] and DBRP[7:0].

Figure 28.17 shows the bit timing chart. **Table 28.40** shows an example of bit timing setting.



Classical CAN mode:
- SS = 1 Tq fixed
- Set TSEG1 to a range of 4 Tq to 16 Tq
- Set TSEG2 to a range of 2 Tq to 8 Tq
- Set SJW to a range of 1 Tq to 4 Tq
- Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 25 Tq
- TSEG1 > TSEG2 ≥ SJW

CANFD mode:
<Nominal bit rate>
- SS = 1 Tq fixed
- Set TSEG1 to a range of 4 Tq to 128 Tq
- Set TSEG2 to a range of 2 Tq to 32 Tq
- Set SJW to a range of 1 Tq to 32 Tq
- Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 161 Tq
- TSEG1 > TSEG2 ≥ SJW

<Data bit rate>
- SS = 1 Tq fixed
- Set TSEG1 to a range of 2 Tq to 16 Tq
- Set TSEG2 to a range of 2 Tq to 8 Tq
- Set SJW to a range of 1 Tq to 8 Tq
- Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 25 Tq
- TSEG1 ≥ TSEG2 ≥ SJW

- SS (synchronization segment):
  The SS is a segment that performs synchronization by monitoring the edge from recessive to dominant bits in the Interframe Space. Interframe Space consists of Intermission, Suspend Transmission, and Bus Idle. All nodes can start transmission during Bus Idle.
- TSEG1 (time segment 1):
  TSEG1 is a segment that absorbs physical delay on the CAN network. The physical delay on the CAN network is twice the total of the delay on the CAN bus, the delay in the input comparator, and the delay in the output driver.
- TSEG2 (time segment 2):
  TSEG2 is a segment that compensates phase error due to an error in frequency.
- SJW (resynchronization jump width)
  The SJW is a length to extend or reduce the time segment to compensate for an error in phase due to phase error.

Figure 28.17 Bit Timing Chart

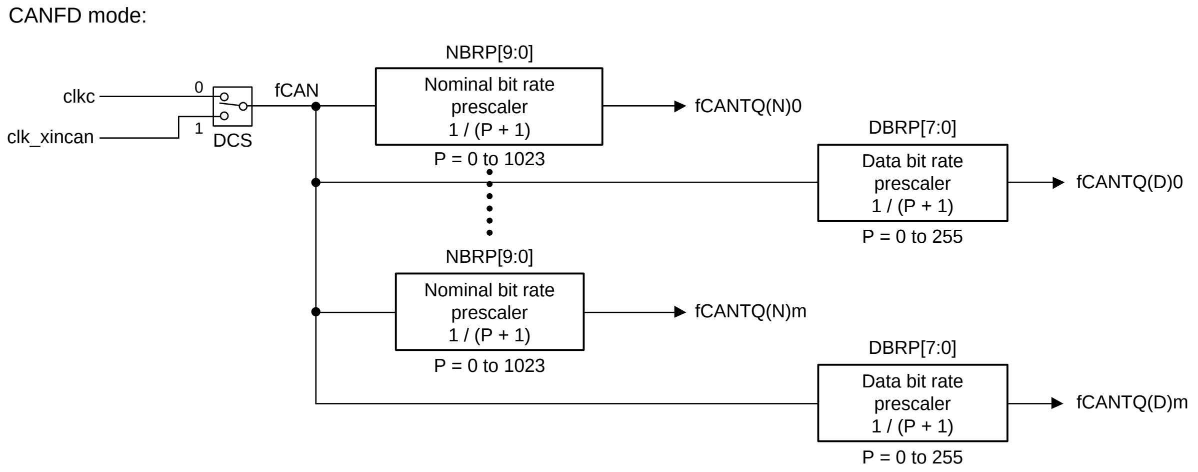
Table 28.40 Example of Bit Timing Setting

1 Bit	Set Value (Tq)				Sample Point (%) *Note*: See **Figure 28.17**.
	SS	TSEG1	TSEG2	SJW	
8 Tq	1	4	3	1	62.50
	1	5	2	1	75.00
10 Tq	1	6	3	1	70.00
	1	7	2	1	80.00
16 Tq	1	10	5	1	68.75
	1	11	4	1	75.00
20 Tq	1	12	7	1	65.00
	1	13	6	1	70.00
50 Tq*[1]	1	39	10	4	80.00

Note 1. Only in CANFD mode

28.11.1.3 Communication Speed Setting

Set the CAN communication speed for each channel using the fCAN, baud rate prescaler division value, and Tq count per bit time. For CANFD mode, set two types of transmission rate (arbitration phase and data phase) for each channel. **Figure 28.18** shows the CAN clock control block diagram, and **Table 28.41** and **Table 28.42** show examples of the communication speed setting.



$$\text{Nominal bit rate} = \frac{\text{fCAN}}{\text{Nominal bit rate prescaler division value} \times (\text{Tq count of 1 nominal bit time})}$$

$$\text{Data bit rate} = \frac{\text{fCAN}}{\text{Data bit rate prescaler division value} \times (\text{Tq count of 1 data bit time})}$$

Note: m = 0, 1
DCS: Bit in the RSCFDn(CFD)GCFG register
BRP[9:0]: Bits in the RSCFDnCmCFG RSCANnCmCFG register
NBRP[9:0]: Bits in the RSCFDnCFDCmNCFG register
DBRP[7:0]: Bits in the RSCFDnCFDCmDCFG register
fCAN: CAN clock
fCANTQm: CANmTq clock
fCANTQ(N)m: CANmTq(N) clock
fCANTQ(D)m: CANmTq(D) clock

Figure 28.18 CAN Clock Control Block Diagram

Table 28.41 Example of Communication Speed Setting (Classical CAN Mode)

fCAN Communication Speed	32 MHz	16 MHz	8 MHz
1 Mbps	8 Tq (4) 16 Tq (2)	8 Tq (2) 16 Tq (1)	8 Tq (1)
500 Kbps	8 Tq (8) 16 Tq (4)	8 Tq (4) 16 Tq (2)	8 Tq (2) 16 Tq (1)
250 Kbps	8 Tq (16) 16 Tq (8)	8 Tq (8) 16 Tq (4)	8 Tq (4) 16 Tq (2)
125 Kbps	8 Tq (32) 16 Tq (16)	8 Tq (16) 16 Tq (8)	8 Tq (8) 16 Tq (4)

Table 28.42 Example of Transmission Rate Setting (Nominal Bit Rate and Data Bit Rate in CANFD Mode)

fCAN Communication Speed	32 MHz	16 MHz
Nominal bit rate 1 Mbps Data bit rate 4 Mbps	Nominal bit rate 32 Tq (1) Data bit rate 8 Tq (1)	None
Nominal bit rate 500 Kbps Data bit rate 2 Mbps	Nominal bit rate 64 Tq (1) Data bit rate 16 Tq (1)	Nominal bit rate 32 Tq (1) Data bit rate 8 Tq (1)

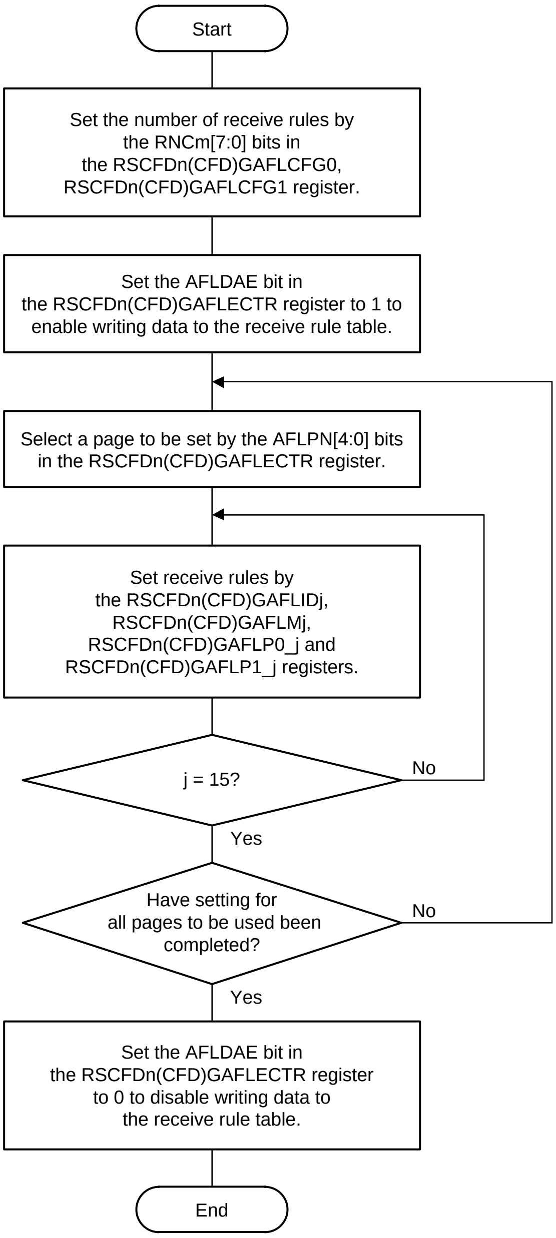
Note: Values in ( ) are baud rate prescaler division values.

28.11.1.4 Receive Rule Setting

Receive rules can be set using receive rule-related registers.

Up to 16 receive rules can be registered per page. Specify pages 0 to 23 (for 6-channel unit) by the AFLPN[4:0] bits in the RSCFDn(CFD)GAFLECTR register. Set receive rule table write enable/disable using the AFLDAE bit.

Figure 28.19 shows the receive rule setting procedure.



Note: m = 0, 1, j = 0 to 15

Figure 28.19 Receive Rule Setting Procedure

28.11.1.5 Buffer Setting

Set the number of buffers to be used (number of messages to be stored) and interrupt sources. Also set the payload storage size for CANFD mode. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

In classical CAN mode, up to 2048 bytes of the RAM can be used in receive buffers and FIFO buffers. Up to 128 buffers are available, and 16 bytes are used per buffer. Configure the buffers so that the following conditions are met.

> Number of receive buffers + total number of depth of receive FIFO buffers x + total number of depth of transmit/receive FIFO buffers k ≤ 128 buffers

In CANFD mode, up to 3584 bytes of the RAM can be used in receive buffers and FIFO buffers. Configure the buffers so that the following conditions are met.

> Number of receive buffers × (12 + payload storage size) + total of (number of depth × (12 + payload storage size)) of receive FIFO buffers x + total of (number of depth × (12 + payload storage size)) of transmit/receive FIFO buffers k ≤ 3584 bytes

Figure 28.20 shows the buffer configuration. **Figure 28.21** shows the buffer setting procedure.

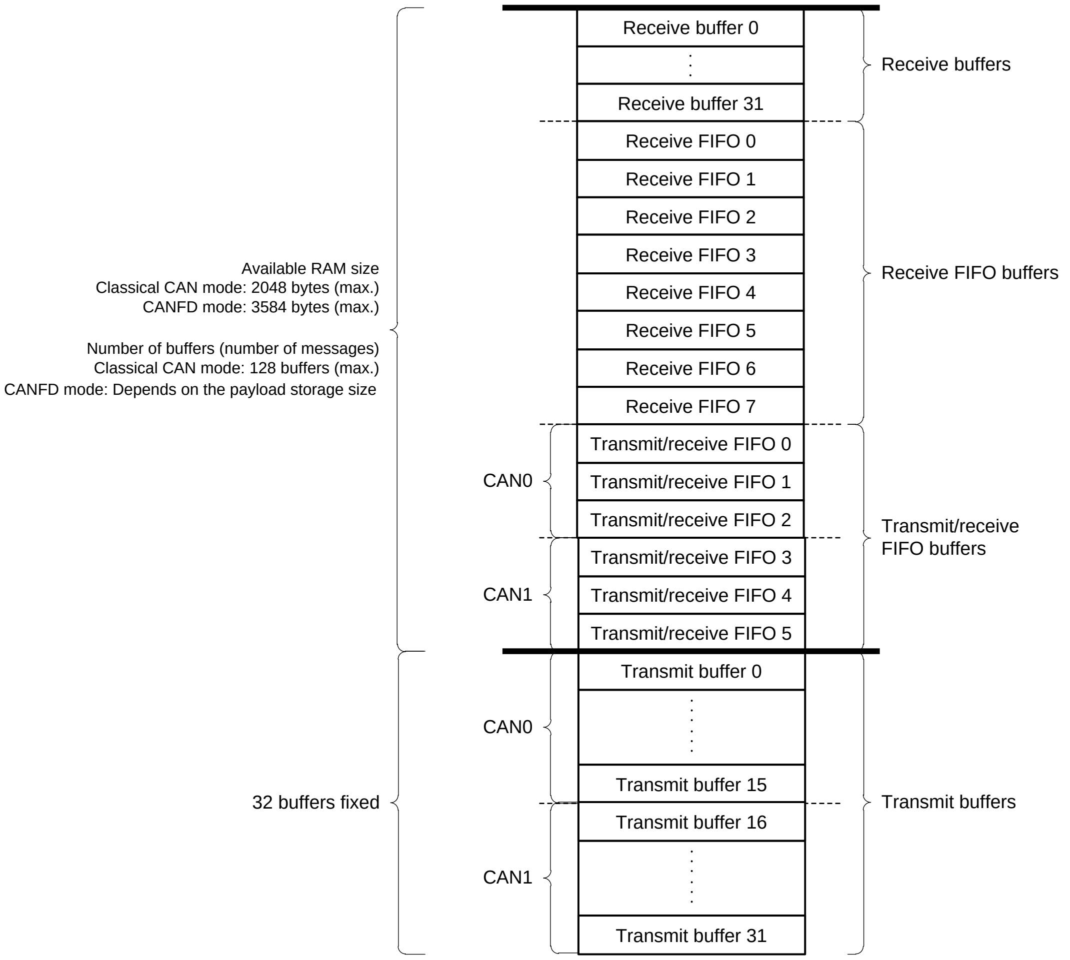
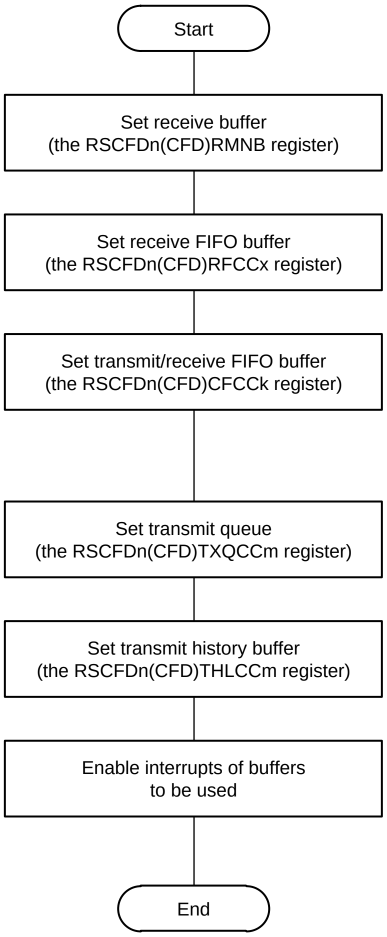


Figure 28.20 Buffer Configuration

NOTE

Receive buffers, receive FIFO buffers, transmit/receive FIFO buffers, and transmit buffers are located in succession.



- Set the number of receive buffers (0 to 32) by the NRXMB[7:0] bits.
- Set the payload storage size of the buffer by the RMPLS[1:0] bits. (Only in CANFD mode)

- Select receive interrupt request timing by the RFIGCV[2:0] bits.
- Select an interrupt source by the RFIM bit.
- Set the number of FIFO buffer stages by the RFDC[2:0] bits.
- Set the payload storage size of the buffer by the RFPLS[2:0] bits. (Only in CANFD mode)

- Set the number of transmit buffer to be linked by the CFTML[3:0] bits.
- Select an interval timer count source by the CFITR and CFITSS bits.
- Select a mode by the CFM[1:0] bits.
- Select receive interrupt request timing by the CFIGCV[2:0] bits.
- Select an interrupt source by the CFIM bit.
- Set the number of FIFO buffer stages by the CFDC[2:0] bits.
- Set the payload storage size of the buffer by the CFPLS[2:0] bits. (Only in CANFD mode)

- Select an interrupt source by the TXQIM bit.
- Set the number of transmit queue stages by the TXQDC[3:0] bits.

- Select a message transmit source buffer type by the THLDTE bit.
- Select an interrupt source by the THLIM bit.

- Enable receive FIFO interrupts by the RFIE bit in RSCFDn(CFD)RFCCx register.
- Enable transmit/receive FIFO transmit interrupts by the CFTXIE bit in RSCFDn(CFD)CFCCk register.
- Enable transmit/receive FIFO receive interrupts by the CFRXIE bit in RSCFDn(CFD)CFCCk register.
- Enable transmit abort interrupts by the TAIE bit in RSCFDn(CFD)CmCTR register.
- Enable transmit complete interrupts by the TMIE bit in RSCFDn(CFD)TMIECy register.
- Enable transmit queue interrupts by the TXQIE bit in RSCFDn(CFD)TXQCCm register.
- Enable transmit history interrupts by the THLIE bit in RSCFDn(CFD)THLCCm register.

Note: k = 0 to 5, x = 0 to 7

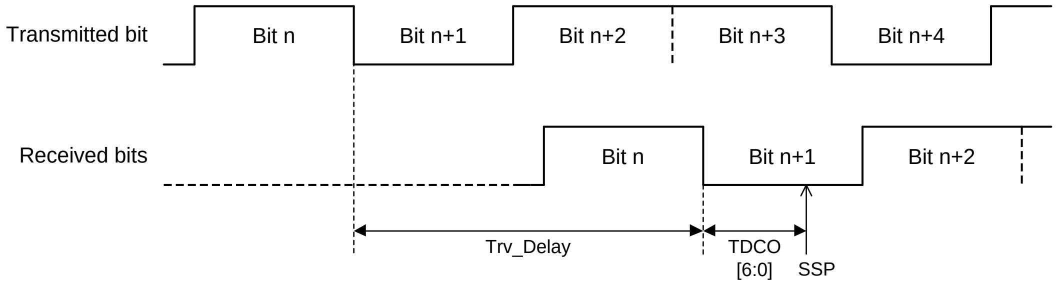
Figure 28.21 Buffer Setting Procedure

28.11.1.6 Transmitter Delay Compensation (Only in CANFD Mode)

A high baud rate is used in CANFD mode. Transmitter delay compensation is provided as a function to accept propagation delay in this case.

To use this function, set the TDCE bit in the RSCFDnCFDCmFDCFG register to 1. Also set the secondary sample point (SSP) timing used in the data phase by the TDCOC bit and TDCO[6:0] bits in the RSCFDnCFDCmFDCFG register.

When the TDCOC bit is 0, the SSP timing equals the total value of the delay measured by the RS-CANFD module and the TDCO[6:0] value. (This value is rounded off to the nearest integer of Tq.) Usually, the TDCO[6:0] value must be equal to SS + TSEG1, the sample point timing.



Note: Bit n: arbitrary bit of data phase
TDCO[6:0]: RSCFDnCFDCmFDCFG register bits

Figure 28.22 SSP Timing

When the TDCOC bit is 1, the SSP timing is determined only by the TDCO[6:0] value. (When the DBRP[7:0] value in the RSCFDnCFDCmDCFG register is larger than 0, the TDCO[6:0] value is also rounded off to the nearest integer of Tq.)

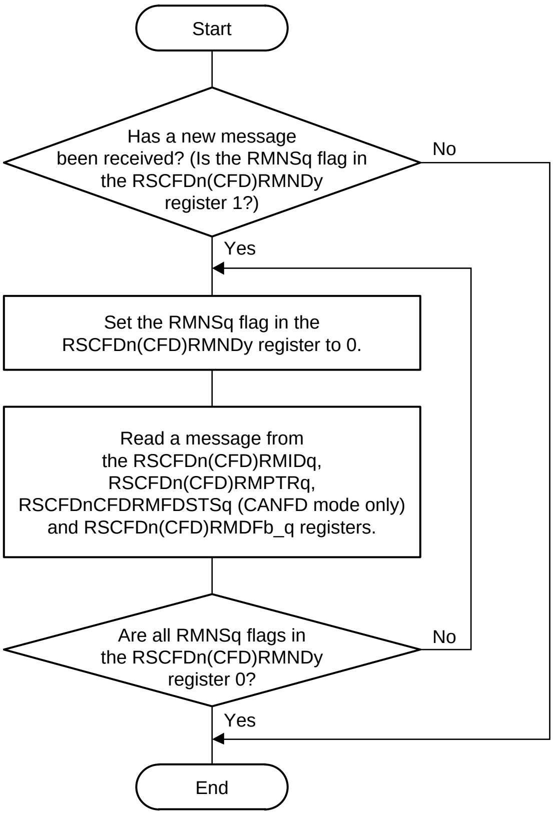
The RS-CANFD module compensates a delay up to (3 CANm bit time − 2 Tq). (Both CANm bit time and Tq are data bit rate values.)

When the TDCE bit in RSCFDnCFDCmFDCFG register is 1 (transmitter delay compensation is enabled), set bits NBRP[9:0] and DBRP[7:0] to the same value that is less than 1.

28.11.2 Reception Procedure

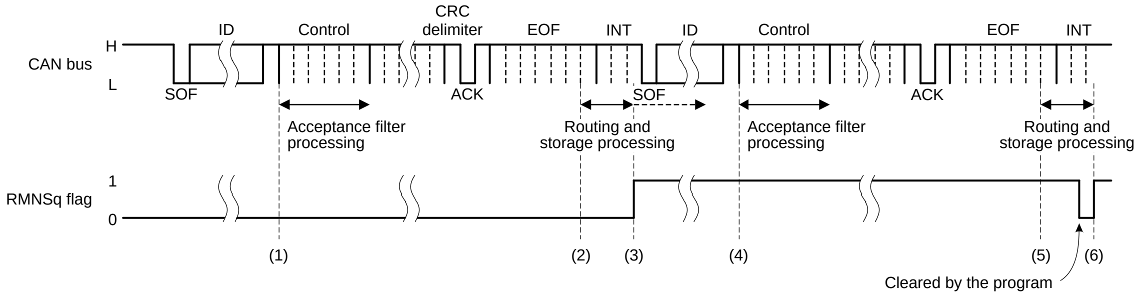
28.11.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCFDn(CFD)RMNDy register (y = 0, q = 0 to 31) is set to 1 (receive buffer q contains a new message). Messages can be read from registers RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq (only in CANFD mode), and RSCFDn(CFD)RMDFb_q (b = 0 or 1 in classical CAN mode, b = 0 to 4 in CANFD mode). **Figure 28.23** shows the receive buffer reading procedure. This procedure ensures the consistency of messages read from registers RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q.



Note: y = 0, q = 0 to 31, b = 0 or 1 (classical CAN mode), 0 to 4 (CANFD mode)

Figure 28.23 Receive Buffer Reading Procedure



Note: y = 0, q = 0 to 31
RMNSq: Flag in the RSCFDn(CFD)RMNDy register

Figure 28.24 Receive Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing, the processing to store the message in the specified receive buffer starts.
When the message storage processing starts, the RMNSq flag in the corresponding RSCFDn(CFD)RMNDy register is set to 1 (the receive buffer contains a new message). If other channels are performing filter processing or transmit priority determination processing, the routing processing and the storage processing may be delayed.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(6) When the corresponding RMNSq flag is cleared to 0 (the receive buffer contains no new message), this flag is set to 1 again when the message storage processing starts. Even if the RMNSq flag remains 1, a new message is overwritten to the receive buffer. The RMNSq flag should not be cleared to 0 during storage of messages.

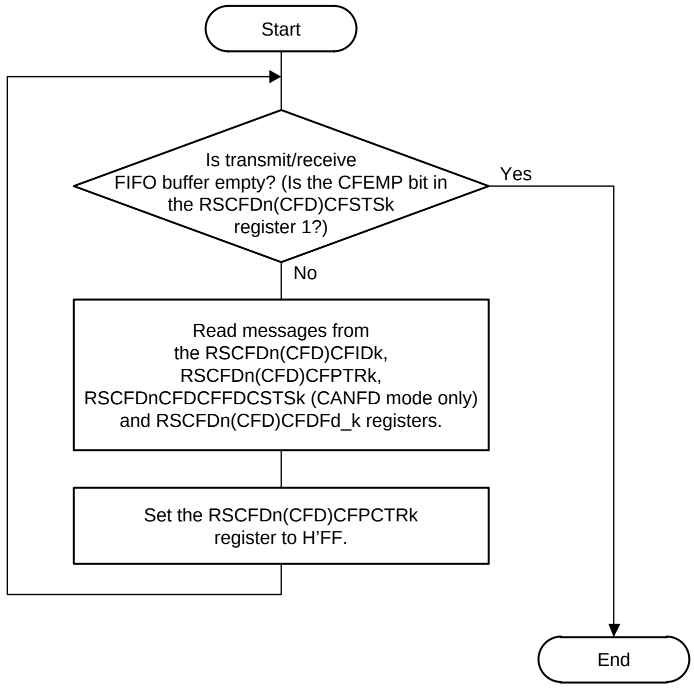
28.11.2.2 FIFO Buffer Reading Procedure

When received messages have been stored in one or more receive FIFO buffers or a transmit/receive FIFO buffer that is set to receive mode or gateway mode, the corresponding message count display counter (RFMC[7:0] bits in the RSCFDn(CFD)RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCFDn(CFD)CFSTSk register (k = 0 to 17)) is incremented. At this time, when the RFIE bit (receive FIFO interrupt is enabled) in the RSCFDn(CFD)RFCCx register or the CFRXIE bit (transmit/receive FIFO receive interrupt is enabled) in the RSCFDn(CFD)CFCCk register is set to 1, an interrupt request is generated. Received messages can be read from the RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx (only in CANFD mode), and RSCFDn(CFD)RFDFd_x (d = 0 or 1 in classical CAN mode, d = 0 to 15 in CANFD mode) registers for receive FIFO buffers, or from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk (only in CANFD mode), and RSCFDn(CFD)CFDFd_k registers for transmit/receive FIFO buffers. Messages in FIFO buffers can be read sequentially on a first-in, first-out basis.

When the message count display counter value matches the FIFO buffer depth (a value set by the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register or the CFDC[2:0] bits in the RSCFDn(CFD)CFCCk register), the RFFLL or CFFLL flag is set to 1 (the receive FIFO buffer is full).

When all messages have been read out of the FIFO buffer, the RFEMP flag in the RSCFDn(CFD)RFSTSx register or the CFEMP flag in the RSCFDn(CFD)CFSTSk register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)).

If the RFE bit or the CFE bit is cleared to 0 (no receive FIFO buffer is used) with the interrupt request flag (RFIF flag in the RSCFDn(CFD)RFSTSx register or CFRXIF flag in the RSCFDn(CFD)CFSTSk register) set to 1 (a receive FIFO interrupt request is present), the interrupt request flag is not automatically cleared to 0. The program must clear the interrupt request flag to 0.



Note: k = 0 to 5, d = 0 or 1 (classical CAN mode), 0 to 15 (CANFD mode)

Figure 28.25 Transmit/Receive FIFO Buffer Reading Procedure

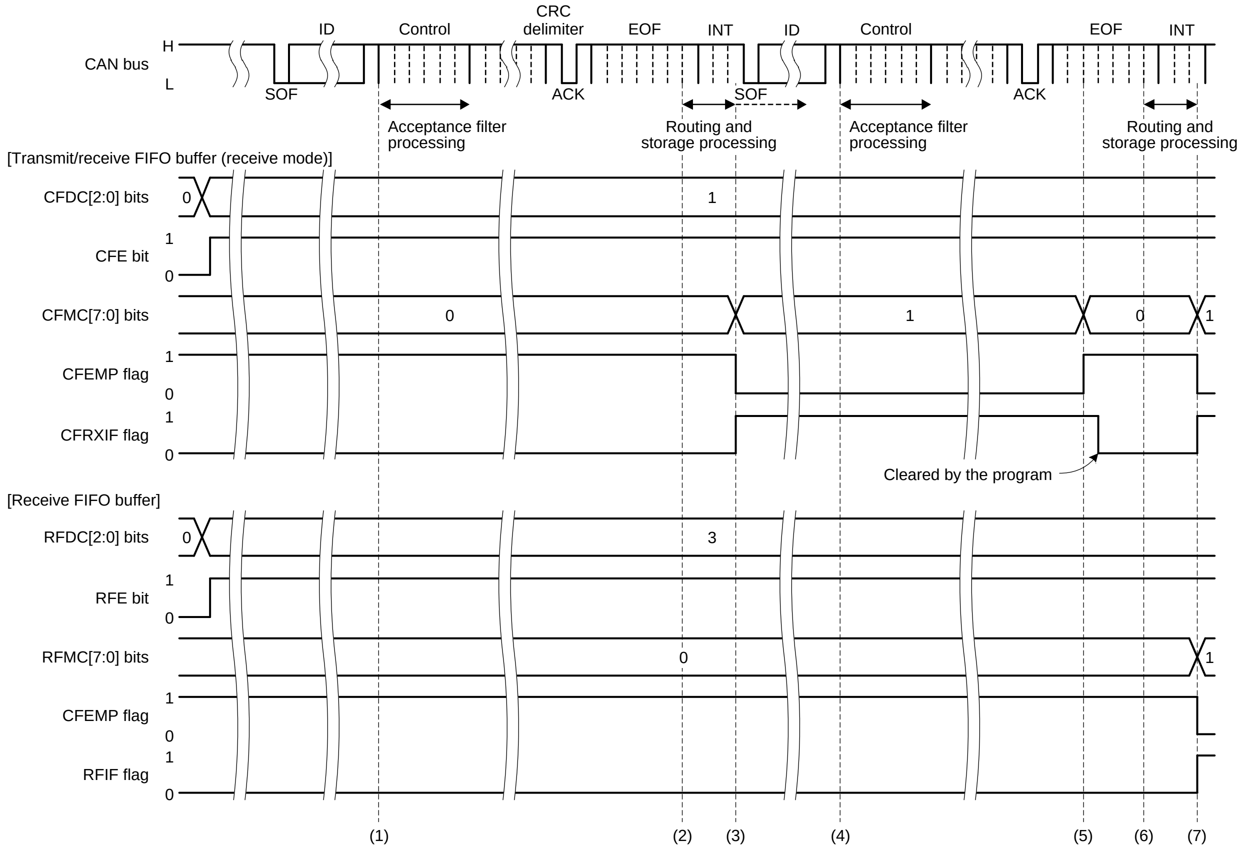
When reading a message in CANFD mode, do not read the RSCFDnCFDRFDFd_x or RSCFDnCFDCFDFd_k register corresponding to the area exceeding the payload storage size specified by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

Table 28.43 Payload Storage Area of Receive FIFO Buffer

Set RFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000b	8 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF1_x
001b	12 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF2_x
010b	16 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF3_x
011b	20 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF4_x
100b	24 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF5_x
101b	32 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF7_x
110b	48 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF11_x
111b	64 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF15_x

Table 28.44 Payload Storage Area of Transmit/Receive FIFO Buffer

Set CFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000b	8 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF1_k
001b	12 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF2_k
010b	16 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF3_k
011b	20 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF4_k
100b	24 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF5_k
101b	32 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF7_k
110b	48 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF11_k
111b	64 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF15_k



Note: CFDC[2:0], CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFRXIF: Flags in the RSCFDn(CFD)CFSTSk register
RFDC[2:0], RFE: Bits in the RSCFDn(CFD)RFCCx register
RFMC[7:0], RFEMP, RFIF: Flags in the RSCFDn(CFD)RFSTSx register
k = 0 to 5, x = 0 to 7

Figure 28.26 FIFO Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing and the CFE bit in the RSCFDn(CFD)CFCCk register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCk register is 001b or more, the message is stored in the transmit/receive FIFO buffer that is set to receive mode. The CFMC[7:0] value in the RSCFDn(CFD)CFSTSk register is incremented and becomes H’01. When the CFIM bit in the RSCFDn(CFD)CFCCk register is set to 1 (a FIFO receive interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCFDn(CFD)CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). The CFRXIF flag can be reset to 0 by the program.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) Read received messages from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, and RSCFDn(CFD)CFDFd_k registers and write H’FF to the RSCFDn(CFD)CFPCTRk register. This causes the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSk register to be decremented. When CFMC[7:0] becomes H’00, the CFEMP flag in the RSCFDn(CFD)CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

(6) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(7) The message is stored in the transmit/receive FIFO buffer set in receive mode when the message has passed through the DLC filter process if the CFE bit is set to 1 (transmit/receive FIFO buffers are used), and the CFDC[2:0] bits are set to 001b or more. The CFMC[7:0] bit value is incremented by 1 to be H’01. When the CFIM bit is set to 1 (an interrupt occurs each time a message has been received), the CFRXIF flag is set to 1 (a transmit/receive FIFO receive interrupt request is present).
The message is stored in the receive FIFO buffer if the RFE bit in the RSCFDn(CFD)RFCCx register is set to 1 (receive FIFO buffers are used), and the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register are set to 001b or more. The RFMC[7:0] bits in the RSCFDn(CFD)RFSTSx register are set to H’01 by being incremented by 1. When the RFIM bit in the RSCFDn(CFD)RFCCx register is set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCFDn(CFD)RFSTSx register is set to 1 (a receive FIFO interrupt request is present).

28.11.2.3 FIFO Buffer Reading Procedure by DMA Transfer

In CANFD mode, the following FIFO buffers can be read by DMA transfer.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k allocated to channel m (k = 3 × m, m = 0, 1)

The DMA enable bit (RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTC register) can be set at any time. However, before setting the DMA enable bit to 1 (to enable DMA transfer requests), set the receive interrupt enable bit (RFIE bit in the RSCFDnCFDRFCCx register or CFRXIE bit in the RSCFDnCFDCFCCk register) of related FIFO buffers to 0 (to disable interrupts). When DMA transfer requests are enabled, do not write a value to the FIFO control register (RSCFDnCFDRFCCx register or RSCFDnCFDCFCCk register).

When an unread message is remaining in a DMA transfer-enabled FIFO buffer, a DMA transfer request trigger is generated. Specify the FIFO access register address for the transfer source address, and adjust the transfer size so that data can be read to the end of the payload storage area with a single trigger. The end of the payload storage area depends on the payload storage size specified by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

After the end of the payload stored in the FIFO buffer has been read, the RFMC[7:0] value in the RSCFDnCFDRFSTSx register or the CFMC[7:0] value in the RSCFDnCFDCFSTSk register is automatically decremented. After that, if an unread message is remaining in the FIFO buffer, a trigger is generated again.

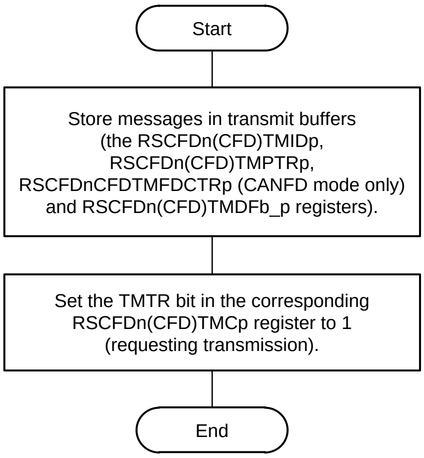
When the RFDMAEx or CFDMAEm bit is set to 0 (to disable DMA transfer requests) during DMA transfer, wait until the DMA transfer status (RFDMASTSx or CFDMASTSm bit in the RSCFDnCFDCDTSTS register) is cleared to 0 (DMA transfer disabled), and then start the next processing (enabling DMA transfer again etc.). When disabling DMA transfer, examine how to process a message remaining in the FIFO buffer and a newly arriving message. When the FIFO buffer is enabled, it continues to receive messages.

28.11.3 Transmission Procedure

28.11.3.1 Procedure for Transmission from Transmit Buffers

Figure 28.27 shows the procedure for transmission from transmit buffers.

Figure 28.28 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission has been successfully completed. **Figure 28.29** shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmit abort has been completed.



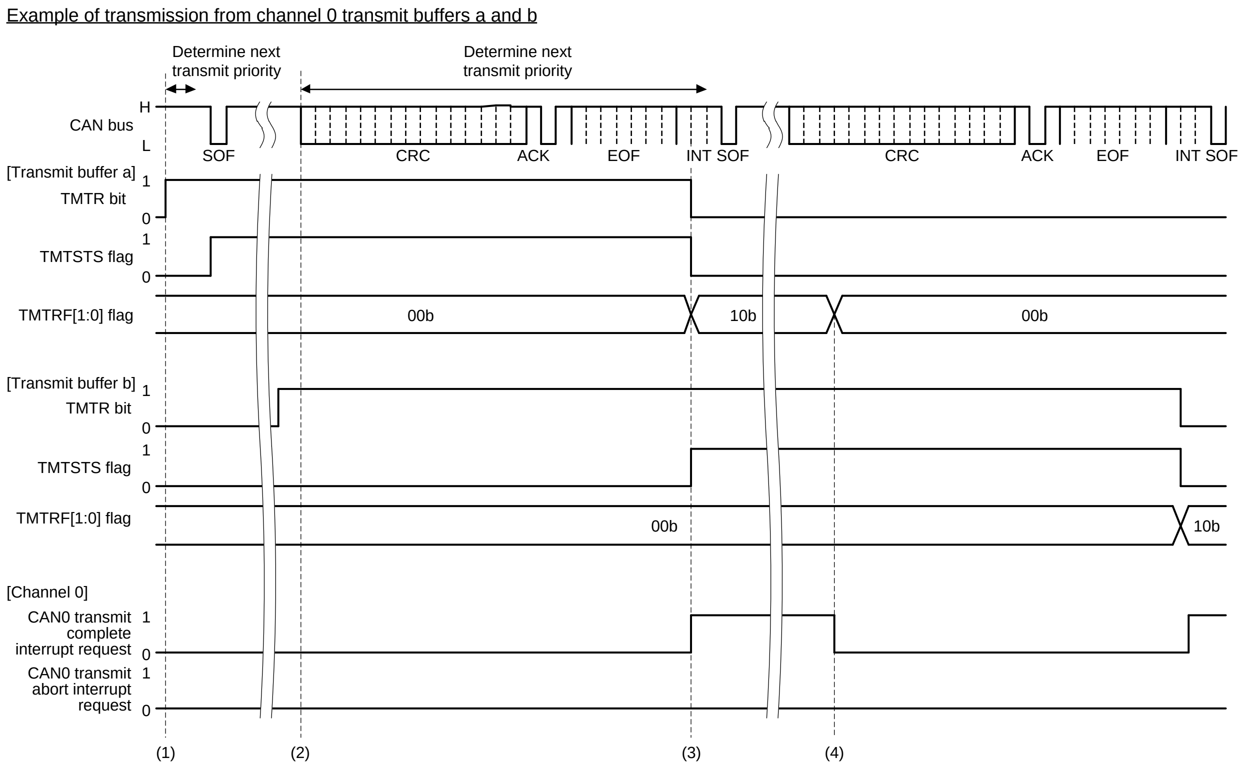
Note: p = 0 to 31, b = 0 or 1 (classical CAN mode), 0 to 4 (CANFD mode)

Figure 28.27 Procedure for Transmission from Transmit Buffers

In CANFD mode and transmit buffer merge mode, messages with a payload size of more than 20 bytes can be transmitted from transmit buffers (16 × m) + 0 and transmit buffers (16 × m) + 3. At this time, transmit buffers (16 × m) + 1 to (16 × m) + 2 and transmit buffers (16 × m) + 4 to (16 × m) + 5 are allocated as a payload storage area. Registers RSCFDnCFDTMIDp, RSCFDnCFDTMPTRp, and RSCFDnCFDTMFDCTRp corresponding to these buffers can be used as data field registers that can store 4-byte data bytes (payload) like the RSCFDnCFDTMDFb_p register. **Table 28.45** shows message storage registers when transmitting a message with a payload size of more than 20 bytes from transmit buffer 0.

Table 28.45 Message Storage Registers in Transmit Buffer Merge Mode (Example of Transmit Buffer 0)

Transmit Buffer	Offset from Base Address	Symbol	Register Function in Transmit Buffer Merge Mode
Transmit buffer 0	H’4000	RSCFDnCFDTMID0	Transmit buffer 0 ID data, transmit history data store enable bit, RTR bit, and IDE bit
	H’4004	RSCFDnCFDTMPTR0	Transmit buffer 0 label data and DLC data
	H’4008	RSCFDnCFDTMFDCTR0	Transmit buffer 0 ESI bit, BRS bit, and FDF bit
	H’400C to H’401C	RSCFDnCFDTMDF0_0 to RSCFDnCFDTMDF4_0	Transmit buffer 0 data bytes 0, 1, 2, and 3 to transmit buffer 0 data bytes 16, 17, 18, and 19
Transmit buffer 1	H’4020	RSCFDnCFDTMID1	Transmit buffer 0 data bytes 20, 21, 22, and 23
	H’4024	RSCFDnCFDTMPTR1	Transmit buffer 0 data bytes 24, 25, 26, and 27
	H’4028	RSCFDnCFDTMFDCTR1	Transmit buffer 0 data bytes 28, 29, 30, and 31
	H’402C to H’403C	RSCFDnCFDTMDF0_1 to RSCFDnCFDTMDF4_1	Transmit buffer 0 data bytes 32, 33, 34, and 35 to transmit buffer 0 data bytes 48, 49, 50, and 51
Transmit buffer 2	H’4040	RSCFDnCFDTMID2	Transmit buffer 0 data bytes 52, 53, 54, and 55
	H’4044	RSCFDnCFDTMPTR2	Transmit buffer 0 data bytes 56, 57, 58, and 59
	H’4048	RSCFDnCFDTMFDCTR2	Transmit buffer 0 data bytes 60, 61, 62, and 63
	H’404C to H’405C	RSCFDnCFDTMDF0_2 to RSCFDnCFDTMDF4_2	Not used



Note: a = 0 to 15, b = 0 to 15
TMTR: Bit in the RSCFDn(CFD)TMCp register
TMTSTS, TMTRF[1:0]: Flags in the RSCFDn(CFD)TMSTSp registers

Figure 28.28 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

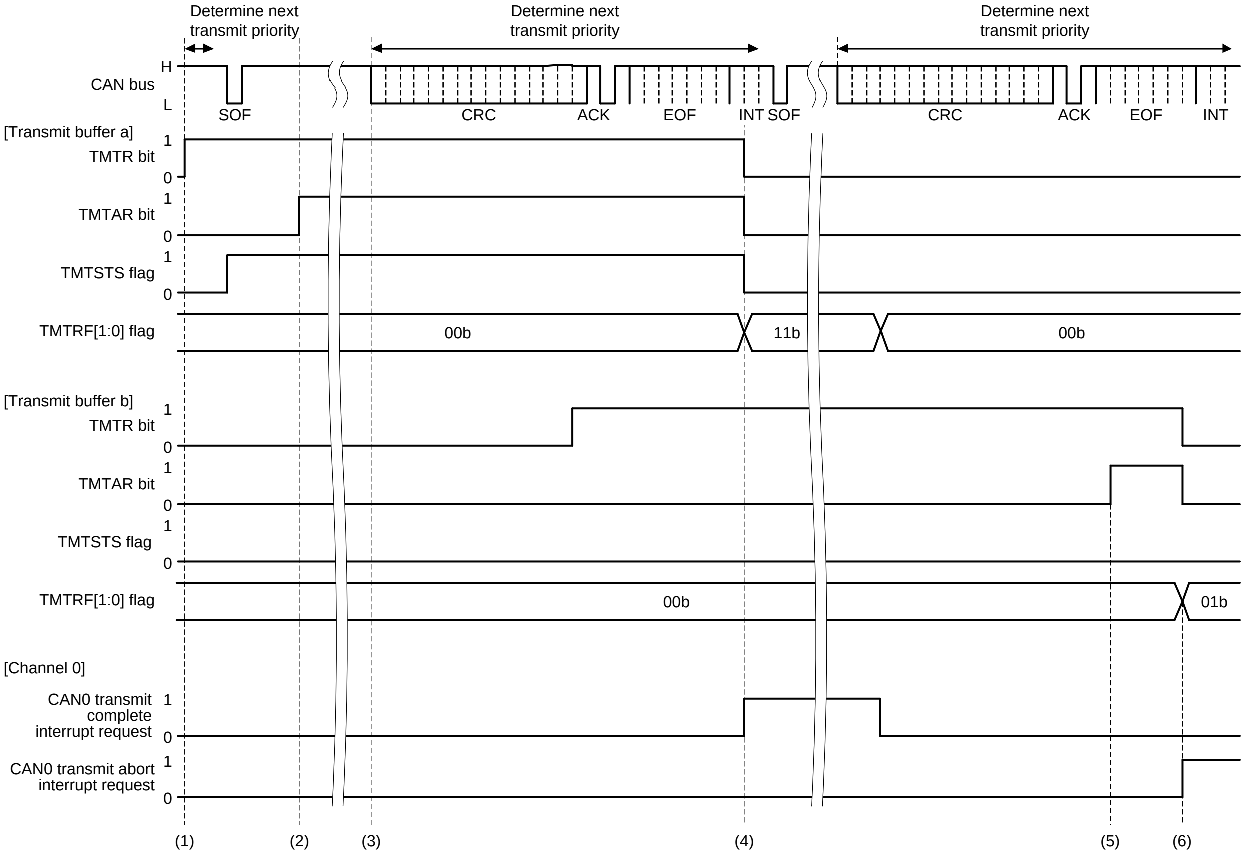
(2) When a transmit request from a buffer is present, the priority determination starts at the first bit of CRC field for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(3) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to 10b (transmission has been completed (without transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMCa register are cleared to 0. When the TMIEa bit in the RSCFDn(CFD)TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00b (transmission is in progress or no transmit request is present).

(4) Before starting the next transmission, set the TMTRF[1:0] flag to 00b. Write the next message to the transmit buffer, and then set the TMTR bit to 1 (transmission is requested). The TMTR bit can be set to 1 only when the TMTRF[1:0] flag value is 00b.

If an arbitration-lost has occurred after transmission is started, the TMTSTS flag is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC field to search the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

Example of transmission from channel 0 transmit buffers a and b



Note: a = 0 to 15, b = 0 to 15

TMTR, TMTAR: Bits in the RSCFDn(CFD)TMCp register

TMTSTS, TMTRF[1:0]: Flags in the RSCFDn(CFD)TMSTSp register

Figure 28.29 Transmit Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When it is determined that the transmit buffer is used for the next transmission or transmission is in progress, message transmission is not aborted unless an error or arbitration loss occurs even if the TMTAR bit is set to 1 (transmit abort is requested).

(3) The priority determination starts at the first bit of the CRC field for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(4) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to 11b (transmission has been completed (with transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMCa register are cleared to 0. When the TMIEa value in the RSCFDn(CFD)TMIEC0 register

is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00b (transmission is in progress or no transmit request is present).

(5) While another CAN node is transmitting data on the CAN bus (TMTSTS flag = 0), if the TMTAR bit is set to 1 while the corresponding channel is determining transmit priority, the TMTR bit cannot be cleared to 0.

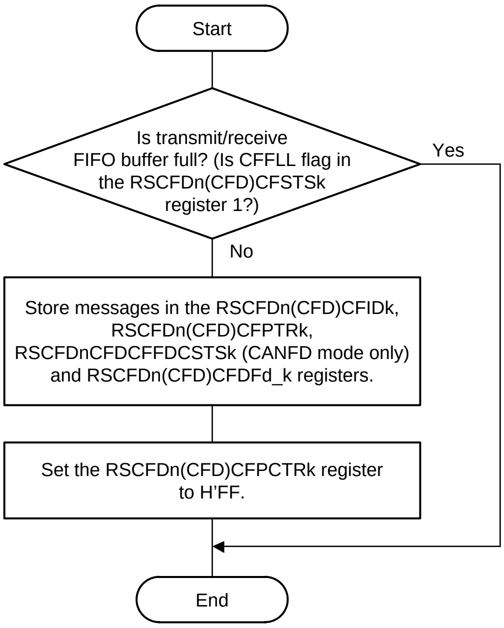
(6) After the internal processing time has passed, the transmission is terminated and the TMTRF[1:0] flag is set to 01b. When the transmit buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not being made, an abort request is immediately accepted and the TMTRF[1:0] flag is set to 01b. At this time, the TMTR and TMTAR bits are cleared to 0. When transmit abort is completed with the TAIE bit in the RSCFDn(CFD)CmCTR register set to 1 (transmit abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00b.

If an arbitration loss has occurred after the CAN channel started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC field to find the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

28.11.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

Figure 28.30 shows the procedure for transmission from transmit/receive FIFO buffers.

Figure 28.31 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission has been successfully completed. **Figure 28.32** shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmit abort has been completed.



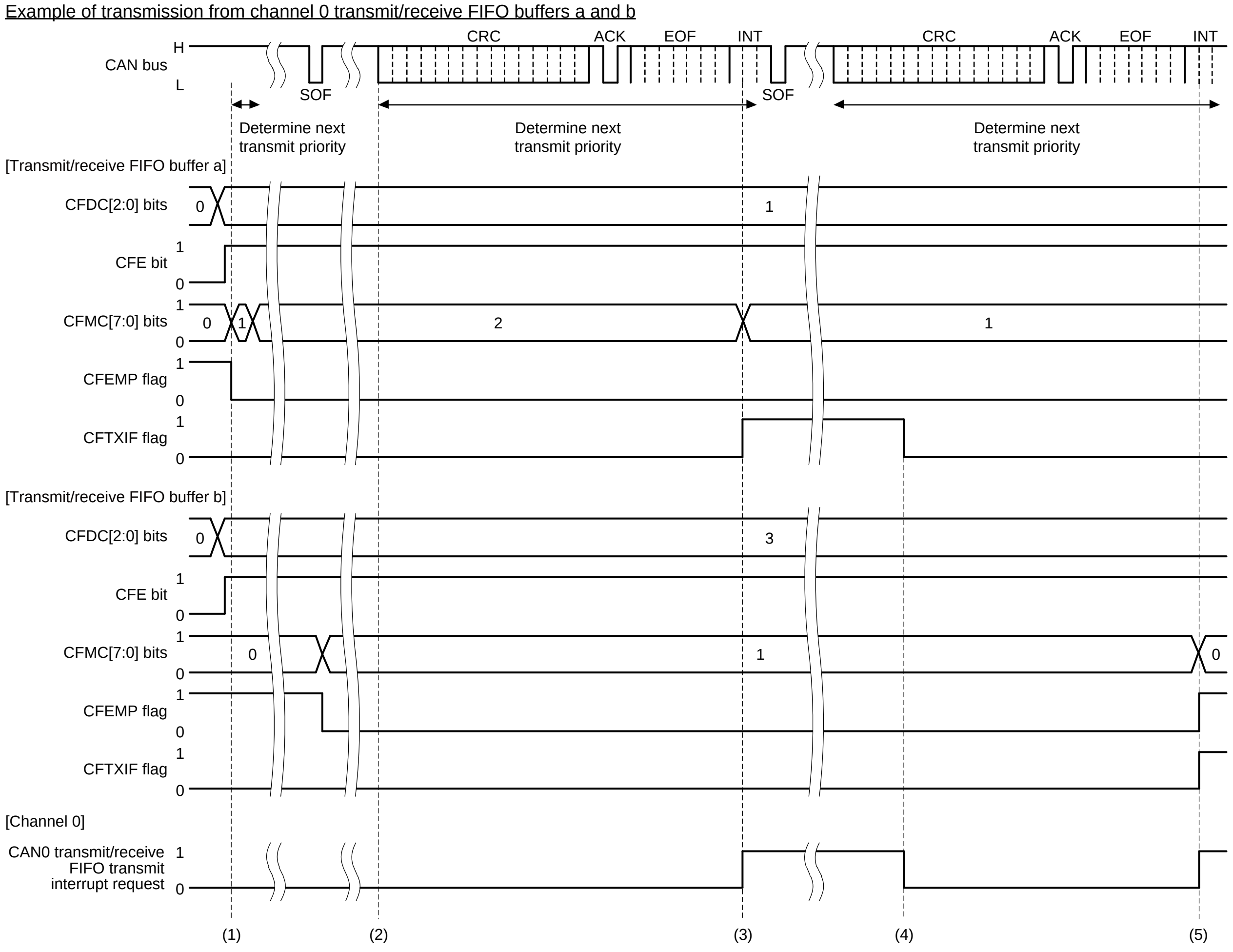
Note: k = 0 to 5, d = 0 or 1 (classical CAN mode), 0 to 15 (CANFD mode)

Figure 28.30 Procedure for Transmission from Transmit/Receive FIFO Buffers

When storing a message, do not write a value to the RSCFDnCFDCFDFd_k register corresponding to the area exceeding the payload storage size specified by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

Table 28.46 Payload Storage Area of Transmit/Receive FIFO Buffer

Set CFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000b	8 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF1_k
001b	12 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF2_k
010b	16 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF3_k
011b	20 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF4_k
100b	24 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF5_k
101b	32 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF7_k
110b	48 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF11_k
111b	64 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF15_k



Note: a = 0 to 2, b = 0 to 2
CFDC[2:0],CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFTXIF: Flags in the RSCFDn(CFD)CFSTSk register

Figure 28.31 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is 001b (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is H’01 or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When a transmit request from a buffer is present, the priority determination starts at the first bit of the CRC field for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

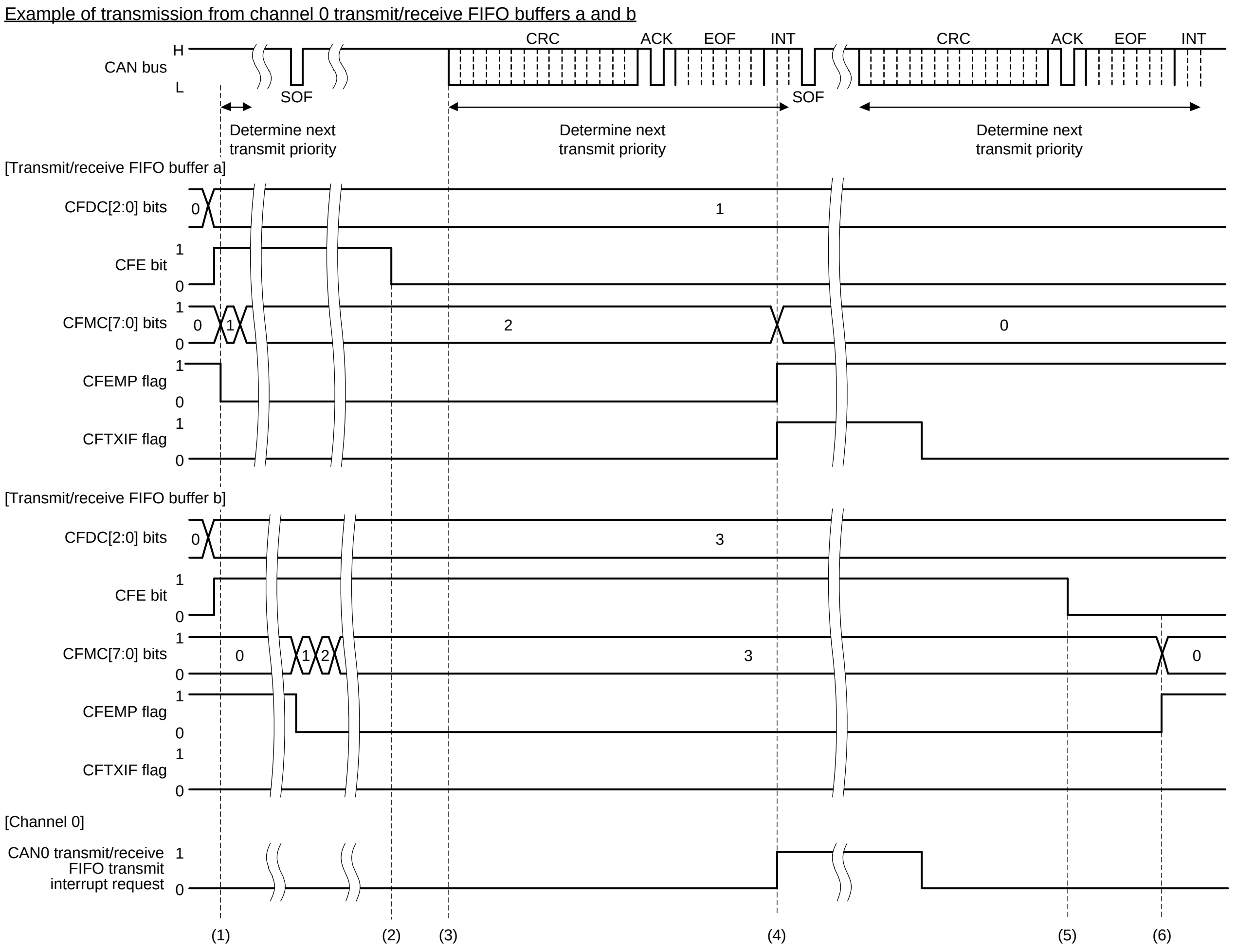
(3) When transmission completes successfully, the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is decremented. Setting the CFIM bit in the RSCFDn(CFD)CFCCa register to 1 (a FIFO transmit interrupt request is

generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSk register to 1 (a transmit/receive FIFO transmit interrupt request is present).

(4) The program can clear the CFTXIF flag.

(5) Message transmission from transmit/receive FIFO buffer b of channel 0 has been completed and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSb register is decremented. The CFMC[7:0] bits are cleared to H’00 and therefore the CFEMP flag in the RSCFDn(CFD)CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

Transmission is continued until the CFEMP flag is set to 1. It is possible to continuously store transmit messages in FIFO buffers until the CFFLL flag in the RSCFDn(CFD)CFSTSa and RSCFDn(CFD)CFSTSb register is set to 1 (the transmit/ receive FIFO buffer is full).



Note: a = 0 to 2, b = 0 to 2
CFDC[2:0], CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFTXIF: Flags in the RSCFDn(CFD)CFSTSk register

Figure 28.32 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is 001b (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is H’01 or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When transmission is in progress or it is determined that the transmit/receive FIFO buffer is used for the next transmission, message transmission is not aborted unless an error or arbitration loss occurs even if the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

(3) When a transmit request from a buffer is present, the priority determination starts at the first bit of the CRC field for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

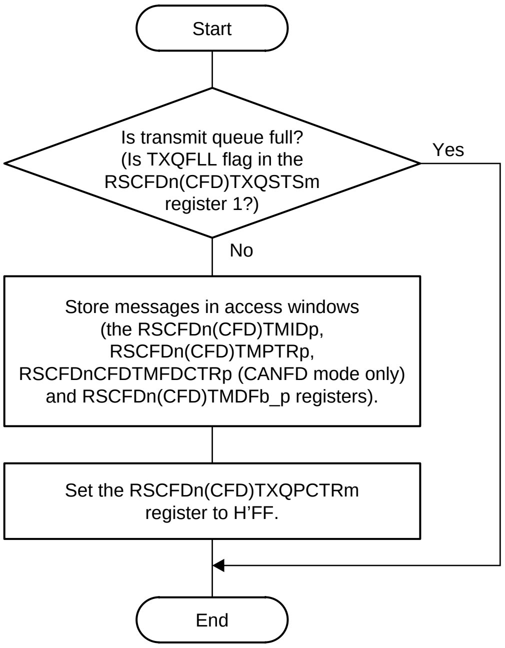
(4) When transmit completes successfully, the CFMC[7:0] value is cleared to H’00. Setting the CFIM bit to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present). The program can clear the CFTXIF flag.

(5) If another CAN node on the CAN bus is transmitting data (not from transmit/receive FIFO buffer b), transmit/receive FIFO buffers cannot be disabled immediately even if the CFE bit in the RSCFDn(CFD)CFCCb register is cleared to 0 (no transmit/receive FIFO buffer is used) during transmit priority determination. (The CFEMP flag in the RSCFDn(CFD)CFSTSb register is not set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) immediately.)

(6) After the internal processing time has passed, transmit/receive FIFO buffers are disabled and the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSb register are cleared to H’00 and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not in progress, the transmit/receive FIFO buffer is immediately disabled. (The CFMC[7:0] bits are cleared to H’00 and the CFEMP flag is set to 1.)

28.11.3.3 Procedure for Transmission from the Transmit Queue

Figure 28.33 shows the procedure for transmission from the transmit queue.

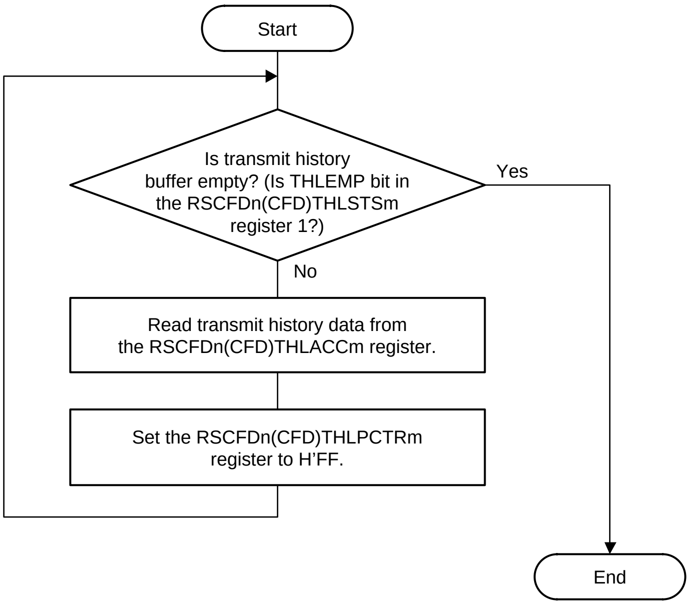


Note: p = m × 16 + 15, m = 0, 1

Figure 28.33 Procedure for Transmission from the Transmit Queue

28.11.3.4 Transmit History Buffer Reading Procedure

Transmit history data can be read from the RSCFDn(CFD)THLACCm register. The next data can be accessed by writing H'FF to the corresponding RSCFDn(CFD)THLPCTRm register (m = 0, 1) after reading a set of data. **Figure 28.34** shows the transmit history buffer reading procedure.



Note: m = 0, 1

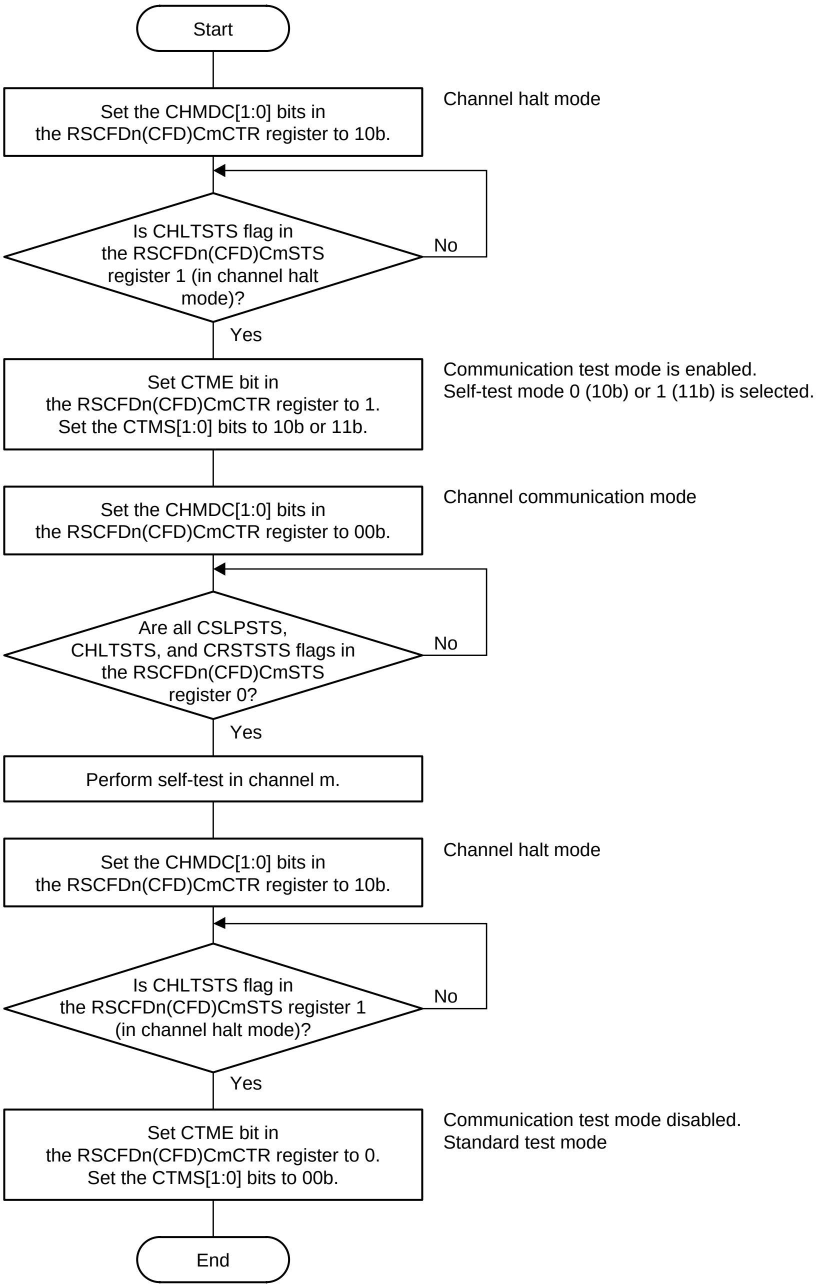
Figure 28.34 Transmit History Buffer Reading Procedure

28.11.4 Test Settings

28.11.4.1 Self-Test Mode Setting Procedure

Self-test mode allows communication test on a channel basis by enabling a CAN node to receive its own transmitted messages.

Figure 28.35 shows the self-test mode setting procedure.



Note: m = 0, 1

Figure 28.35 Self-Test Mode Setting Procedure

28.11.4.2 Procedure for Releasing the Protection

Since the global test function in **Table 28.47** is protected, write the protection release data 1 and release data 2 in succession to the LOCK[15:0] bits in the RSCFDn(CFD)GLOCKK register, then set the target test bit to 1.

Table 28.47 Protection Release Data for Test Function

Test Function	Protection Release Data 1	Protection Release Data 2	Target Bit
RAM test	H’7575	H’8A8A	RTME bit in the RSCFDn(CFD)GTSTCTR register

If an incorrect value is written to the LOCK[15:0] bits, restart from writing the protection release data 1. **Figure 28.36** shows the procedure for releasing the protection.

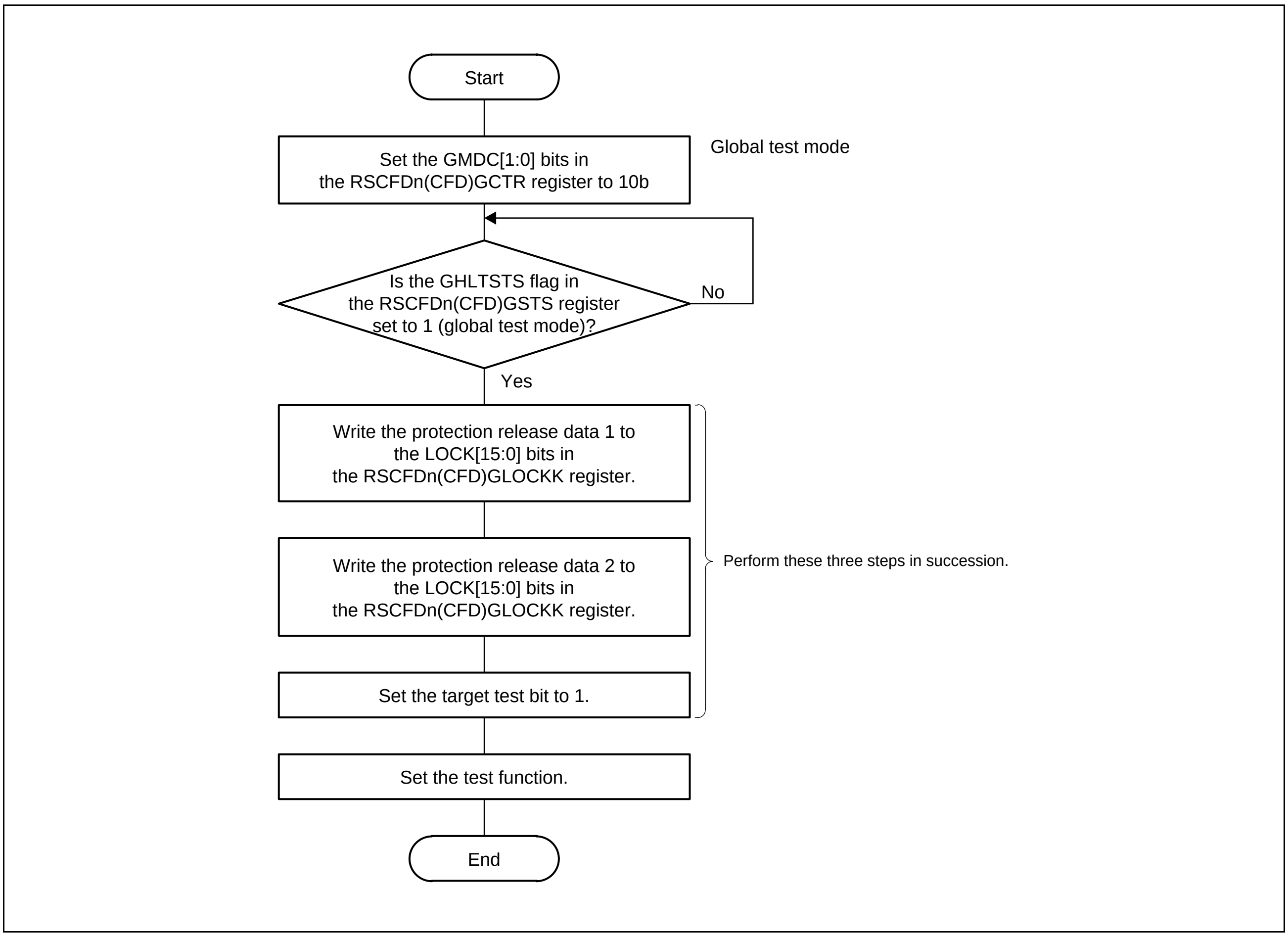
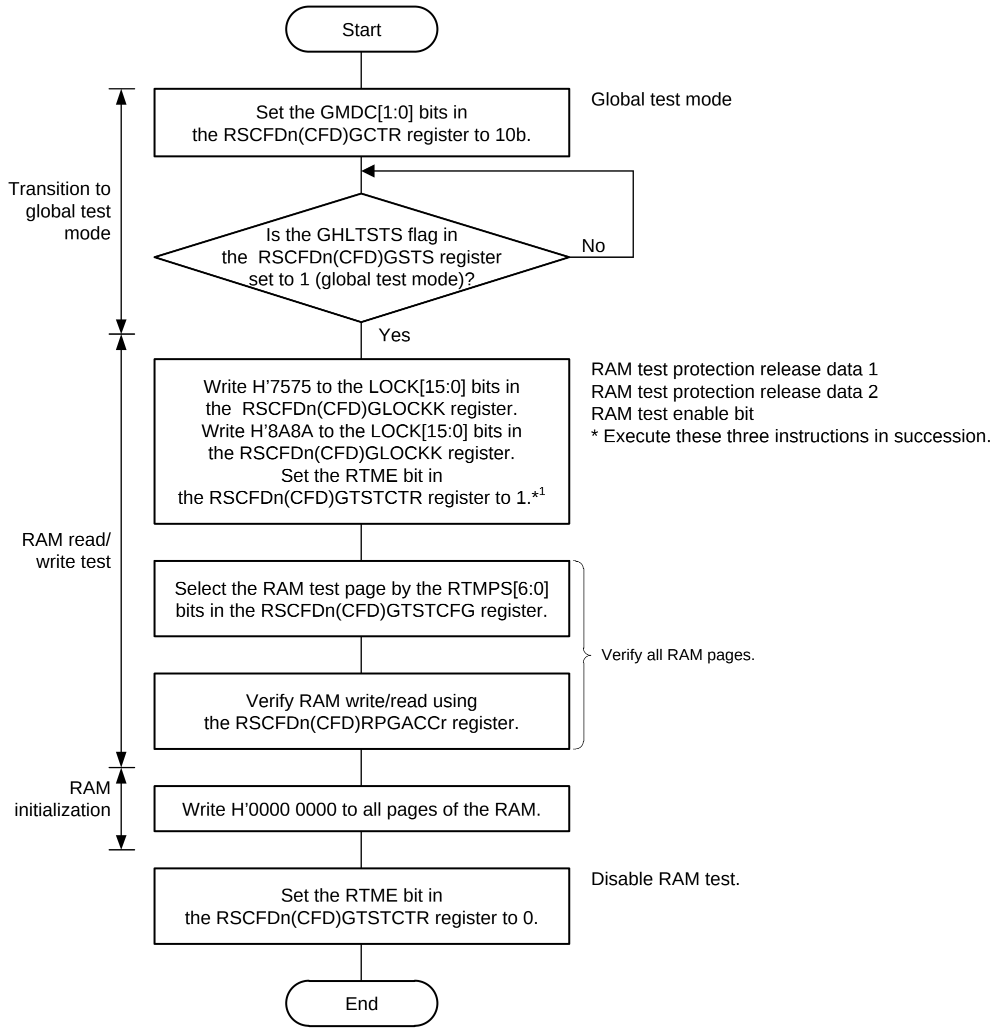


Figure 28.36 Protection Release Procedure

28.11.4.3 RAM Test Setting Procedure

RAM tests include CAN RAM read/write test. The read/write test verifies that data written to the RAM is read correctly. Before closing the RAM test, write H’0000 0000 to all pages of the CAN RAM.

Figure 28.37 shows the RAM test setting procedure.



Note: r = 0 to 63

Note 1. Before transitioning to the RAM test, perform the following.
- Cancel transmission request.
- Disable of each FIFO and transmission queue
- Clear flag of receive buffer

Figure 28.37 RAM Test Setting Procedure

28.11.4.4 Inter-Channel Communication Test Setting Procedure

Communication testing can be performed by transmitting and receiving data between different channels.

Figure 28.38 shows the inter-channel communication test setting procedure.

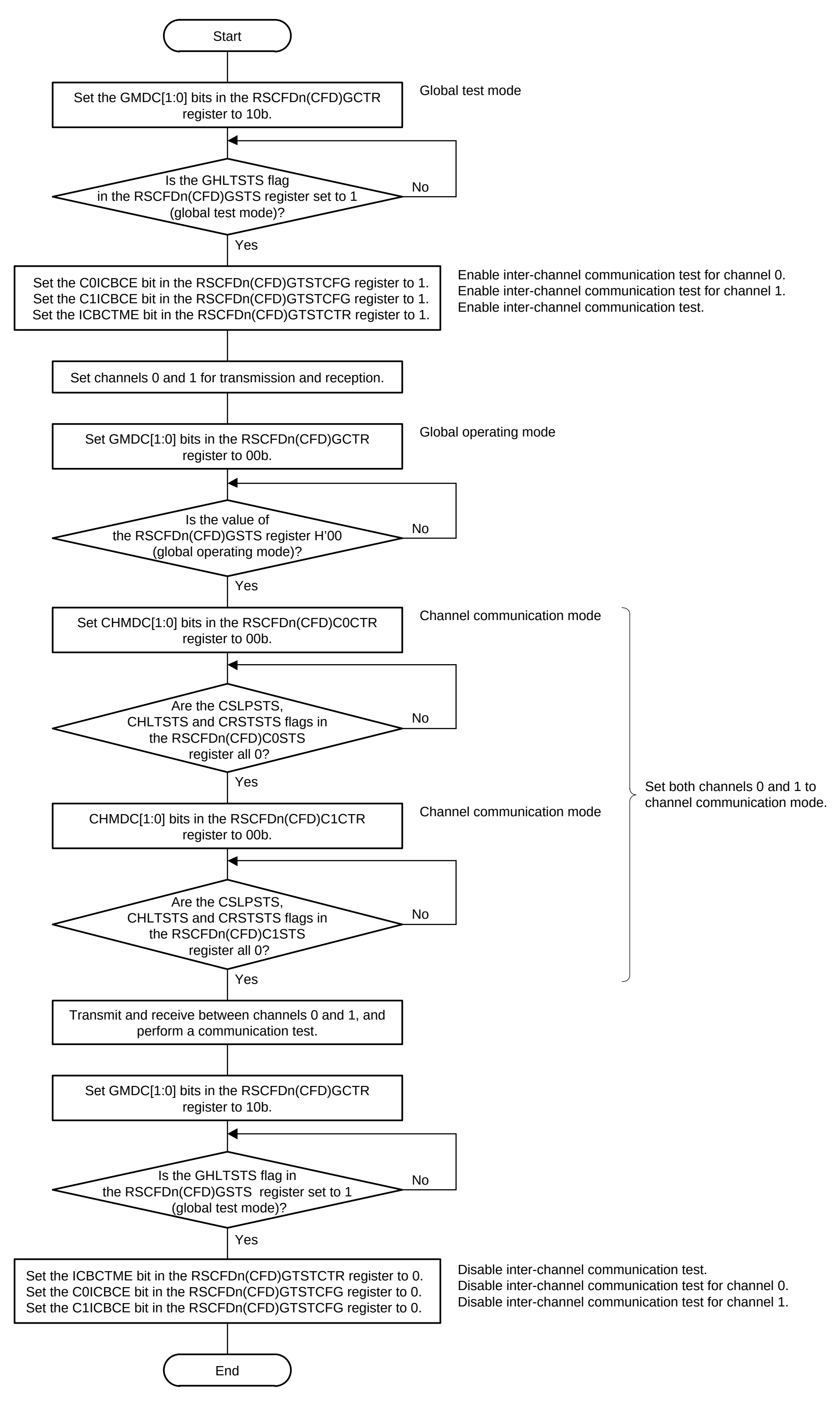


Figure 28.38 Inter-Channel Communication Test Setting Procedure (Example of Communication Test between Channel 0 and Channel 1)

28.12 Notes on the RS-CANFD Module

- When changing interface mode without resetting the RS-CANFD, write the value after reset to all registers and bits that are not allocated to the register map after change and then modify the RSCFDnCFDGRMCFG register.
- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCFDn(CFD)GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCFDn(CFD)CmSTS register (m = 0, 1) for transitions.
- When only classical CAN frames are used in CANFD mode, set the RSCFDnCFDCmDCFG register to the value equal to the set RSCFDnCFDCmNCFG register value.
- The acceptance filter processing checks receive rules sequentially in ascending order from the minimum rule number. If the same ID, IDE bit, or RTR bit value is set for multiple receive rules, the minimum number of receive rule is used for the acceptance filter processing. If the message does not pass through the subsequent DLC filter processing, the data processing is terminated without returning to the acceptance filter processing and the message is not stored in the buffer.
- When linking transmit buffers to transmit/receive FIFO buffers or allocating transmit buffers to transmit queues, set the control register (RSCFDn(CFD)TMCp) of the corresponding transmit buffer to H’00. The status register (RSCFDn(CFD)TMSTSp) of the corresponding transmit buffer should not be used. Flags in other status registers (registers RSCFDn(CFD)TMTRSTS0 to RSCFDn(CFD)TMTRSTS2, RSCFDn(CFD)TMTARSTS0 to RSCFDn(CFD)TMTARSTS2, RSCFDn(CFD)TMTCSTS0 to RSCFDn(CFD)TMTCSTS2, and RSCFDn(CFD)TMTASTS0 to RSCFDn(CFD)TMTASTS2), which correspond to transmit buffers linked to transmit/receive FIFO buffers or allocated to transmit queues remain unchanged. Set the enable bit in the corresponding interrupt enable register (registers RSCFDn(CFD)TMIEC0 to RSCFDn(CFD)TMIEC2) to 0 (transmit buffer interrupt is disabled).
- When using transmit buffer merge mode (in CANFD mode), write H’00 to the control register (RSCFDn(CFD)TMCp) of the transmit buffer corresponding to the transmit buffer allocated as a payload storage area. Set the enable bit of corresponding interrupt enable registers (RSCFDn(CFD)TMIEC0 to RSCFDn(CFD)TMIEC2) to 0 (to disable interrupts).
- Transmit buffers that are linked to transmit/receive FIFO buffers must not bet allocated to transmit queues. Do not allocate a transmit buffer allocated as a payload storage area in transmit buffer merge mode (in CANFD mode) to the transmit queue either.
- Only a single transmit/receive FIFO buffer can be linked to a transmit buffer. Do not link two or more transmit/receive FIFO buffers to transmit buffers of the same number.
- When the CANm bit time clock is selected as a timestamp counter clock source, the timestamp counter stops when the corresponding channel has transitioned to channel reset mode or channel halt mode.
- In case of an attempt to store a new received message when the receive FIFO buffer and the transmit/receive FIFO buffer are full, the new message is discarded. If you wish to store a new transmit message in the transmit/receive FIFO buffer or the transmit queue, check that the transmit/receive FIFO buffer or the transmit queue is not full.
- In the case of registers that access the RAM, the value after reset shown in **Section 28.3, Registers (Classical CAN Mode)** and **Section 28.4, Registers (CANFD Mode)** indicate the values cleared by initialization of the CAN RAM. Values before clear are undefined. The following registers apply.
  - Receive rule (RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, RSCFDn(CFD)GAFLP1_j registers)
  - Receive buffers (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, RSCFDn(CFD)RMDFb_q registers)

  - Receive FIFO buffer access registers (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx, and RSCFDn(CFD)RFDFd_x registers)
  - Transmit/receive FIFO buffer access registers (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDn(CFD)CFDFd_k registers)
  - Transmit buffers (RSCFDn(CFD)TMIDp, RSCFDn(CFD)TMPTRp, RSCFDnCFDTMFDCTRp, and RSCFDn(CFD)TMDFb_p registers)
  - Transmit history access register (RSCFDn(CFD)THLACCm registers)
  - RAM test page access register (RSCFDn(CFD)RPGACCr registers)
- The values of unused receive buffers (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q registers), receive FIFO buffer access registers (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx, and RSCFDn(CFD)RFDFd_x registers) and transmit/receive FIFO buffer access registers (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDn(CFD)CFDFd_k registers) are undefined when the RS-CANFD module transitions to global operation mode or global test mode after exiting from global reset mode.

29. Sampling Rate Converter (SRC)

The sampling rate converter (SRC) handles sampling rate conversion.

29.1 Features

The following lists the features of the SRC.

■ Data format

16-bit stereo/16-bit monauralJan 31, 2024

■ Sampling rates

- Input:
  Selectable from 8 kHz, 11.025 kHz, 12 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, and 48 kHz
- Output:
  Selectable from 8 kHz*, 16 kHz*, 32 kHz, 44.1 kHz, and 48 kHz

Note: * This frequency is only available when the input sampling rate is 44.1 kHz.

■ Processing capacity

A sample output interval is a minimum of 4.62 μs (462 clock cycles at Pϕ = 100 MHz).

■ SNR

80 db or higher

■ Five interrupt sources

Input data FIFO buffer empty, output data FIFO buffer full, output data FIFO buffer overwrite, output data FIFO buffer underflow, and conversion end

■ Two DMA transfer sources

Input data FIFO buffer empty and output data FIFO buffer full

Figure 29.1 shows the configuration of the SRC.

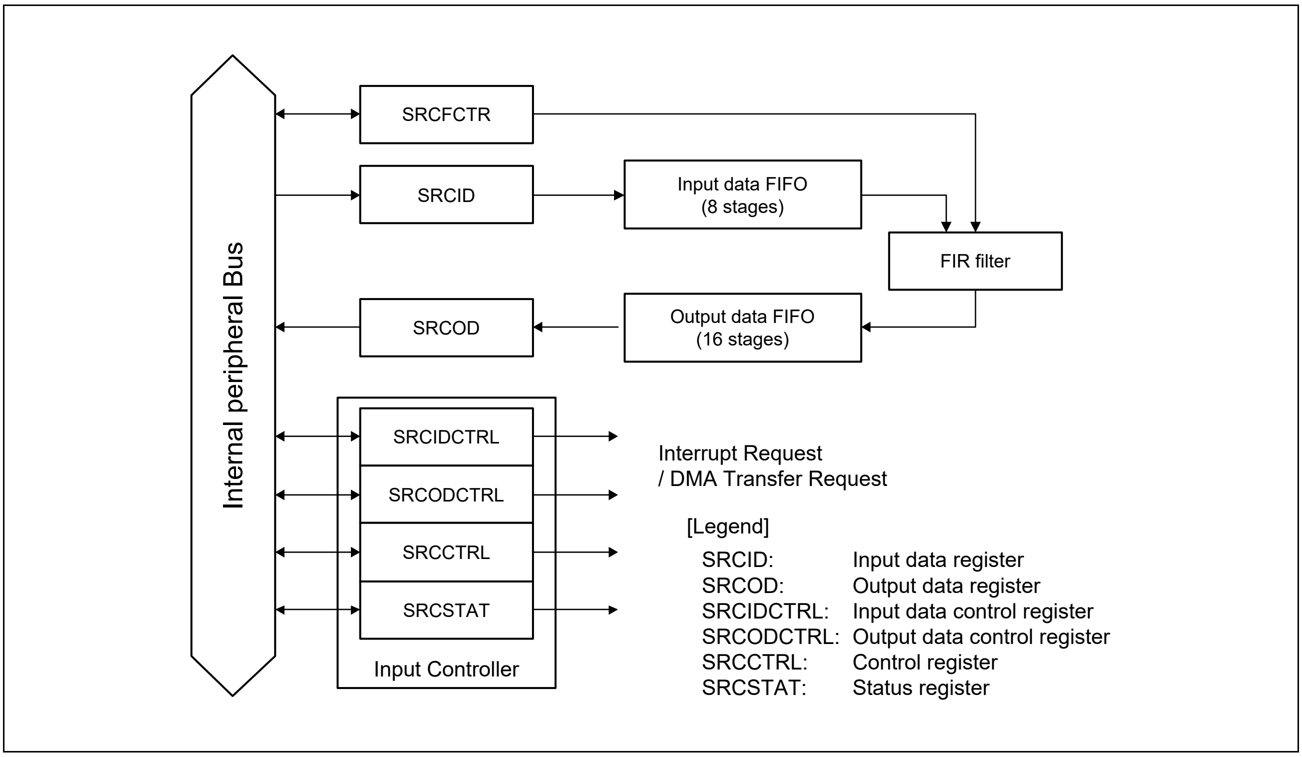


Figure 29.1 Block Diagram of SRC

29.2 Register Configuration

Table 29.1 shows the register configuration. The SRC address space is offset from the base address. The base address of SRC is as follows.

SRC base address: H’0_1004_0000 (Overall Address Space)

SRC base address: H’5004_0000 (Cortex-M33 Address Space Secure)

SRC base address: H’4004_0000 (Cortex-M33 Address Space Non-Secure)

Table 29.1 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Higher-order bits in the input data register	SRCID*3	R/W	H’0000	H’7000	16
Lower-order bits in the input data register			H’0000	H’7002	16
Higher-order bits in the output data register	SRCOD*4	R	H’0000	H’7004	16
Lower-order bits in the output data register			H’0000	H’7006	16
Input data control register	SRCIDCTRL	R/W	H’0000	H’7008	16
Output data control register	SRCODCTRL	R/W	H’0000	H’700A	16
Control register	SRCCTRL	R/W	H’0000	H’700C	16
Status register	SRCSTAT	R/(W)*1	H’0002	H’700E	16
Filter coefficient tables	SRCFCTRs	R/W	Undefined	H’0000 to H’56BF	32*2

Note 1. Bits 15 to 6 and 4 are read-only. Only 0 can be written to bits 5, 3, 1 and 0 after having read as 1.

Note 2. The actual width of read and written data is 22 bits.

Note 3. For the input data register (SRCID), the 16 higher-order bits and the 16 lower-order bits must be written in that order.

Note 4. For the output data register (SRCOD), the 16 higher-order bits and the 16 lower-order bits must be read in that order.

29.3 Register Descriptions

29.3.1 Input Data Register (SRCID)

SRCID is a 32-bit readable and writable register that is used to input the data before sampling rate conversion. All the bits are read as 0. The data input to SRCID is stored in the 8-stage input data FIFO buffer. When the number of data units in the input data FIFO buffer is 8, writing to SRCID has no effect.

For stereo data, bits 31 to 16 are for Lch data, and bits 15 to 0 are for Rch data. For monaural data, data in bits 31 to 16 is valid, and data in bits 15 to 0 is invalid.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The data subject to sampling rate conversion is aligned differently depending on the IED bit setting in SRCIDCTRL. **Table 29.2** shows the relationship between the IED bit setting and data alignment.

Table 29.2 Alignment of Data before Sampling Rate Conversion

IED	Left channel [15:8]	Left channel [7:0]	Right channel [15:8]*1	Right channel [7:0]*1
0	SRCID[31:24]	SRCID[23:16]	SRCID[15:8]	SRCID[7:0]
1	SRCID[23:16]	SRCID[31:24]	SRCID[7:0]	SRCID[15:8]

Note 1. When processing monaural data, the data in these bits is invalid.

29.3.2 Output Data Register (SRCOD)

SRCOD is a 32-bit read-only register used to output the data after sampling rate conversion. The data in the output data FIFO buffer is read through SRCOD. When the number of data units in the 16-stage output data FIFO buffer is zero after the start of conversion, the value previously read is read again.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

The data in SRCOD is aligned differently depending on the OCH and OED bit setting in SRCODCTRL. **Table 29.3** shows the correspondence between the OCH and OED bit setting and data alignment in SRCOD.

Table 29.3 Alignment of Data in SRCOD

OCH	OED	SRCOD[31:24]	SRCOD[23:16]	SRCOD[15:7]	SRCOD[7:0]
0	0	Left channel [15:8]	Left channel [7:0]	Right channel [15:8][*2]	Right channel [7:0][*2]
	1	Left channel [7:0]	Left channel [15:8]	Right channel [7:0][*2]	Right channel [15:8][*2]
1[*1]	0	Right channel [15:8]	Right channel [7:0]	Left channel [15:8]	Left channel [7:0]
	1	Right channel [7:0]	Right channel [15:8]	Left channel [7:0]	Left channel [15:8]

Note 1. When processing monaural data, do not set the bit to 1.

Note 2. When processing monaural data, the data in these bits is invalid.

29.3.3 Input Data Control Register (SRCIDCTRL)

SRCIDCTRL is a 16-bit readable and writable register that specifies the endian format of input data, enables or disables the interrupt requests, and specifies the triggering number of data units.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IED	IEN	—	—	—	—	—	—	IFTRG[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	IED	0	R/W	Input Data Endian Specifies the endian format of the input data. 0: Little endian 1: Big endian *Note*: This bit must be rewritten while the setting of the SRCEN bit is 0.
8	IEN	0	R/W	Input Data FIFO Buffer Empty Interrupt Enable Enables or disables the input data FIFO buffer empty interrupt request to be issued when the number of data units in the input FIFO buffer becomes equal to or smaller than the triggering number specified by the IFTRG1 and IFTRG0 bits, thus resulting in the IINT bit in the status register (SRCSTAT) being set to 1. 0: Input data FIFO buffer empty interrupt is disabled. 1: Input data FIFO buffer empty interrupt is enabled.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	IFTRG[1:0]	00	R/W	Input FIFO Buffer Data Triggering Number Specifies the condition in terms of the number on which the IINT bit in the status register (SRCSTAT) is set to 1. When the number of data units in the input FIFO buffer becomes equal to or smaller than the triggering number listed below, the IINT bit is set to 1. 00: 0 01: 2 10: 4 11: 6

29.3.4 Output Data Control Register (SRCODCTRL)

SRCODCTRL is a 16-bit readable and writable register that specifies whether to exchange the channels for the output data, specifies the endian format of output data, enables or disables the interrupt requests, and specifies the triggering number of data units.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OCH	OED	OEN	—	—	—	—	—	—	OFTRG[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	OCH	0	R/W	Output Data Channel Exchange Specifies whether to exchange the channels for the output data register (SRCOD). 0: Does not exchange the channels (the same order as data input) 1: Exchanges the channels (the opposite order from data input) *Note 1.* When processing monaural data, do not set this bit to 1. *Note 2.* This bit must be rewritten while the setting of the SRCEN bit is 0.
9	OED	0	R/W	Output Data Endian Specifies the endian format of the output data. 0: Little endian 1: Big endian *Note*: This bit must be rewritten while the setting of the SRCEN bit is 0.
8	OEN	0	R/W	Output Data FIFO Buffer Full Interrupt Enable Enables or disables the output data FIFO buffer full interrupt request to be issued when the number of data units in the output FIFO buffer becomes equal to or greater than the number specified by the OFTRG1 and OFTRG0 bits, thus resulting in the OINT bit in the status register (SRCSTAT) being set to 1. 0: Output data FIFO buffer full interrupt is disabled. 1: Output data FIFO buffer full interrupt is enabled.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	OFTRG[1:0]	00	R/W	Output FIFO Buffer Data Trigger Number Specifies the condition in terms of the number on which the OINT bit in the status register (SRCSTAT) is set to 1. When the number of data units in the output FIFO buffer becomes equal to or greater than the number listed below, the OINT bit is set to 1. 00: 1 01: 4 10: 8 11: 12

29.3.5 Control Register (SRCCTRL)

SRCCTRL is a 16-bit readable and writable register that enables or disables access to the filter coefficient tables, enables or disables module operation, enables or disables the interrupt requests, specifies processing to flush or clear the internal working memory, and sets the input and output sampling rates.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FICRAE	—	CEEN	SRCEN	UDEN	OVEN	FL	CL	IFS[3:0]				—	OFS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	FICRAE	0	R/W	Access to the Filter Coefficient Tables Enable Enables or disables reading from and writing to the filter coefficient table RAMs. 0: Disables reading from and writing to the filter coefficient table RAMs. 1: Enables reading from and writing to the filter coefficient table RAMs. *Note*: This bit must be rewritten while the setting of the SRCEN bit is 0.
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	CEEN	0	R/W	Conversion End Interrupt Enable Enables or disables the conversion end interrupt to be generated when the CEF bit in SRCSTAT is set to 1 after flush processing is completed and all the output data is read. 0: Disables conversion end interrupt requests. 1: Enables conversion end interrupt requests. *Note*: For details of flush processing, see the description of the FL bit in this register.
12	SRCEN	0	R/W	Module Enable Enables or disables this module operation. Writing 1 while SRCEN = 0 clears the internal working memory. 0: Disables this module operation. 1: Enables this module operation. *Note*: When SRCEN = 1, do not change the settings of the following bits.
11	UDEN	0	R/W	Output Data FIFO Buffer Underflow Interrupt Enable Enables or disables the output data FIFO buffer underflow interrupt to be generated when output data FIFO buffer is read and the UDF bit in SRCSTAT is set to 1 while the number of data units in the output data FIFO buffer is zero. 0: Disables output data FIFO buffer underflow interrupt requests. 1: Enables output data FIFO buffer underflow interrupt requests.

Register	Bit	Bit Name
SRCIDCTRL	9	IED
SRCODCTRL	10, 9	OCH, OED
SRCCTRL	15, 7 to 4, 2 to 0	FICRAE, IFS[3:0], OFS[2:0]

Bit	Bit Name	Initial Value	R/W	Description
10	OVEN	0	R/W	Output Data FIFO Buffer Overwrite Interrupt Enable Enables or disables the output data FIFO buffer overwrite interrupt request to be issued when the conversion for the next data has been completed while the number of data units in the output FIFO buffer is eight, thus setting the OVF bit in the status register (SRCSTAT) to 1. 0: Output data FIFO buffer overwrite interrupt is disabled. 1: Output data FIFO buffer overwrite interrupt is enabled. ● When OVEN = 1: Conversion processing is stopped until the OVF bit is cleared by the CPU accessing to SRCSTAT when the output data FIFO buffer overwrite interrupt is generated. At this time, conversion result writing to the output data FIFO buffer is also stopped. ● OVEN = 0: The OVF bit is automatically cleared when the output data FIFO buffer has space, and conversion processing can be continued.
9	FL	0	R/W	Internal Working Memory Flush Writing 1 to this bit starts converting the sampling rate of all the data in the input FIFO buffer, input buffer memory, and intermediate memory (i.e., flush processing). After all external input data are eliminated, sampling rate conversion processing proceeds for the data left in the internal memory. This bit is always read as 0. When SRCEN = 0, writing 1 to this bit does not trigger flush processing. In addition, when 1 is written to the FL bit while the number of data units in the input buffer memory is less than the values shown in **Table 29.6**, valid output data cannot be received. Thus the internal working memory is cleared without triggering the flush processing.
8	CL	0	R/W	Internal Working Memory Clear Writing 1 to this bit clears the input FIFO buffer, output FIFO buffer, input buffer memory, intermediate memory, and accumulator, after which this bit is cleared to 0. This bit is always read as 0. Even when SRCEN = 0, writing 1 to this bit clears the processing.
7 to 4	IFS[3:0]	0000	R/W	Input Sampling Rate Specifies the input sampling rate. 0000: 8.0 kHz 0001: 11.025 kHz 0010: 12.0 kHz 0011: Setting prohibited 0100: 16.0 kHz 0101: 22.05 kHz 0110: 24.0 kHz 0111: Setting prohibited 1000: 32.0 kHz 1001: 44.1 kHz 1010: 48.0 kHz 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
3	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	OFS[2:0]	000	R/W	Output Sampling Rate Specifies the output sampling rate. 000: 44.1 kHz 001: 48.0 kHz 010: 32.0 kHz 011: Setting prohibited 100: 8.0 kHz*1 101: 16.0 kHz*1 110: Setting prohibited 111: Setting prohibited *Note 1.* This setting of OFS[2:0] is only valid when the setting of IFS[3:0] is 1001.

After flush processing has been completed, the number of output data units obtained as a result of conversion can be calculated by using the following formula.

$$\frac{\text{Number of output data units} - 1}{\text{Output sampling rate}} = \frac{\text{Number of input data units} \times \text{n} - 1}{\text{Input sampling rate} \times \text{n}}$$

$$\text{Number of output data units} = \left\lfloor (\text{Number of input data units} \times \text{n} - 1) \times \frac{\text{Output sampling rate}}{\text{Input sampling rate} \times \text{n}} \right\rfloor + 1$$

Table 29.4 Values of the n in the Formula Above

OFS Setting (Output Sampling Rate [kHz])	IFS Setting (Input Sampling Rate [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000 (44.1)	6	4	4	3	2	2	3	—	1
001 (48.0)	6	4	4	3	2	2	3	1	—
010 (32.0)	4	8	4	2	4	2	—	2	1
100 (8.0)	—	—	—	—	—	—	—	1	—
101 (16.0)	—	—	—	—	—	—	—	1	—

Conversion processing by this module does not start and thus output data are not obtained until the specified number of data units have been input. The minimum number of input data units required for both obtaining the first output data and performing flush processing depends on the IFS and OFS bit settings.

Table 29.5 and **Table 29.6** show the relation between the settings of the IFS and OFS bits and the number of input data units required.

Table 29.5 Relation between Sampling Rate Settings and Number of Initial Input Data Units Required

OFS Setting (Output Sampling Rate [kHz])	IFS Setting (Input Sampling Rate [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000 (44.1)	38	40	40	43	48	48	43	—	63
001 (48.0)	38	40	40	43	48	48	43	32	—
010 (32.0)	40	37	40	48	40	48	—	48	63
100 (8.0)	—	—	—	—	—	—	—	63	—
101 (16.0)	—	—	—	—	—	—	—	63	—

Table 29.6 Relation between Sampling Rate Settings and Number of Input Data Units Required for Flush Processing

OFS Setting (Output Sampling Rate [kHz])	IFS Setting (Input Sampling Rate [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000 (44.1)	27	24	24	22	16	16	22	—	1
001 (48.0)	27	24	24	22	16	16	22	32	—
010 (32.0)	24	29	24	16	24	16	—	16	1
100 (8.0)	—	—	—	—	—	—	—	1	—
101 (16.0)	—	—	—	—	—	—	—	1	—

29.3.6 Status Register (SRCSTAT)

SRCSTAT is a 16-bit readable and writable register that indicates the number of data units in the input and output data FIFO buffers, whether the various interrupt sources have been generated or not, and the flush processing status.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFDN[4:0]					IFDN[3:0]				—	CEF	FLF	UDF	OVF	IINT	OINT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R/(W)*1	R	R/(W)*1	R/(W)*1	R/(W)1*	R/(W)1*

Note 1. Only 0 can be written after having read as 1.

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	OFDN[4:0]	00000	R	Output FIFO Buffer Data Count Indicates the number of data units in the output FIFO buffer.
10 to 7	IFDN[3:0]	0000	R	Input FIFO Buffer Data Count Indicates the number of data units in the input FIFO buffer.
6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	CEF	0	R/(W)*1	Conversion End Flag Indicates that all the output data is read after flush processing is completed. [Clearing conditions] ● 0 has been written to the CEF bit after reading CEF = 1 ● 1 has been written to the CL bit in SRCCTRL ● 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] ● The number of data units in the output data FIFO buffer is zero on completion of flush processing
4	FLF	0	R	Flush Processing Status Flag Indicates whether flush processing is in progress or not. [Clearing conditions] ● Flush processing has been completed ● 1 has been written to the CL bit in SRCCTRL ● 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] ● 1 has been written to the FL bit in SRCCTRL Note that this bit will never be set when flush processing is not performed.
3	UDF	0	R/(W)*1	Output FIFO Buffer Underflow Interrupt Request Flag Indicates that the output data FIFO buffer is read when the number of data units in the output data FIFO buffer is zero. [Clearing conditions] ● 0 has been written to the UDF bit after reading UVF = 1 ● 1 has been written to the CL bit in SRCCTRL ● 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] ● The output data FIFO buffer is read while the number of data units in the output FIFO buffer is zero

Bit	Bit Name	Initial Value	R/W	Description
2	OVF	0	R/(W)[*1]	Output Data FIFO Buffer Overwrite Interrupt Request Flag Indicates that the sampling rate conversion for the next data has been completed when the output data FIFO buffer is full. The conversion is stopped until the OVF flag is cleared. [Clearing conditions] ● 0 has been written to the OVF bit after reading OVF = 1 while the OVEN bit in SRCCTRL is 1 ● The number of data units in the output FIFO buffer decreases after reading SRCOD while the OVEN bit in SRCCTRL is 0, that is, empty space is available in the output FIFO buffer ● 1 has been written to the CL bit in SRCCTRL ● 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] ● The sampling rate conversion for the next data has been completed when the output FIFO buffer is full
1	IINT	1	R/(W)[*1]	Input Data FIFO Buffer Empty Interrupt Request Flag Indicates that the number of data units in the input FIFO buffer has become equal to or smaller than the triggering number specified by the IFTRG[1:0] bits in the input data control register (SRCIDCTRL). [Clearing conditions] ● 0 has been written to the IINT bit after reading IINT = 1 ● Handling of the last transfer by DMA transfer [Setting conditions] ● The number of data units in the input FIFO buffer has become equal to or smaller than the specified triggering number ● 1 has been written to the CL bit in SRCCTRL ● 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0
0	OINT	0	R/(W)[*1]	Output Data FIFO Buffer Full Interrupt Request Flag Indicates that the number of data units in the output FIFO buffer has become equal to or greater than the triggering number specified by the OFTRG[1:0] bits in the output data control register (SRCODCTRL). [Clearing conditions] ● 0 has been written to the OINT bit after reading OINT = 1 ● Handling of the last transfer by DMA transfer ● 1 has been written to the CL bit in SRCCTRL ● 1 has been written to the SRCEN bit in SRCCTRL while SRCEN is 0 [Setting condition] ● The number of data units in the output FIFO buffer has become equal to or greater than the specified triggering number

Note 1. Only 0 can be written after having read as 1.

29.3.7 Filter Coefficient Tables (SRCFCTRs)

SRCFCTRs are 32-bit readable and writable random access memory (RAMs) that hold the filter coefficients to be used in the sampling rate conversion processing. The peripheral bus only has access to the RAMs when the FICRAE and SRCEN bits in the control register (SRCCTRL) are set to 1 and 0, respectively. Bits 31 to 22 are reserved and always read as 0. Bits 21 to 0 are used as the register that holds filter coefficient values, of which the initial values are undefined.

Each filter coefficient table can only hold 22 bits × 5552 words of coefficient data. The sampling rate conversion operation and its characteristics are only guaranteed when the separately provided coefficient data are used.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	SRCFCOE[21:16]					
Initial Value	0	0	0	0	0	0	0	0	0	0	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCFCOE[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

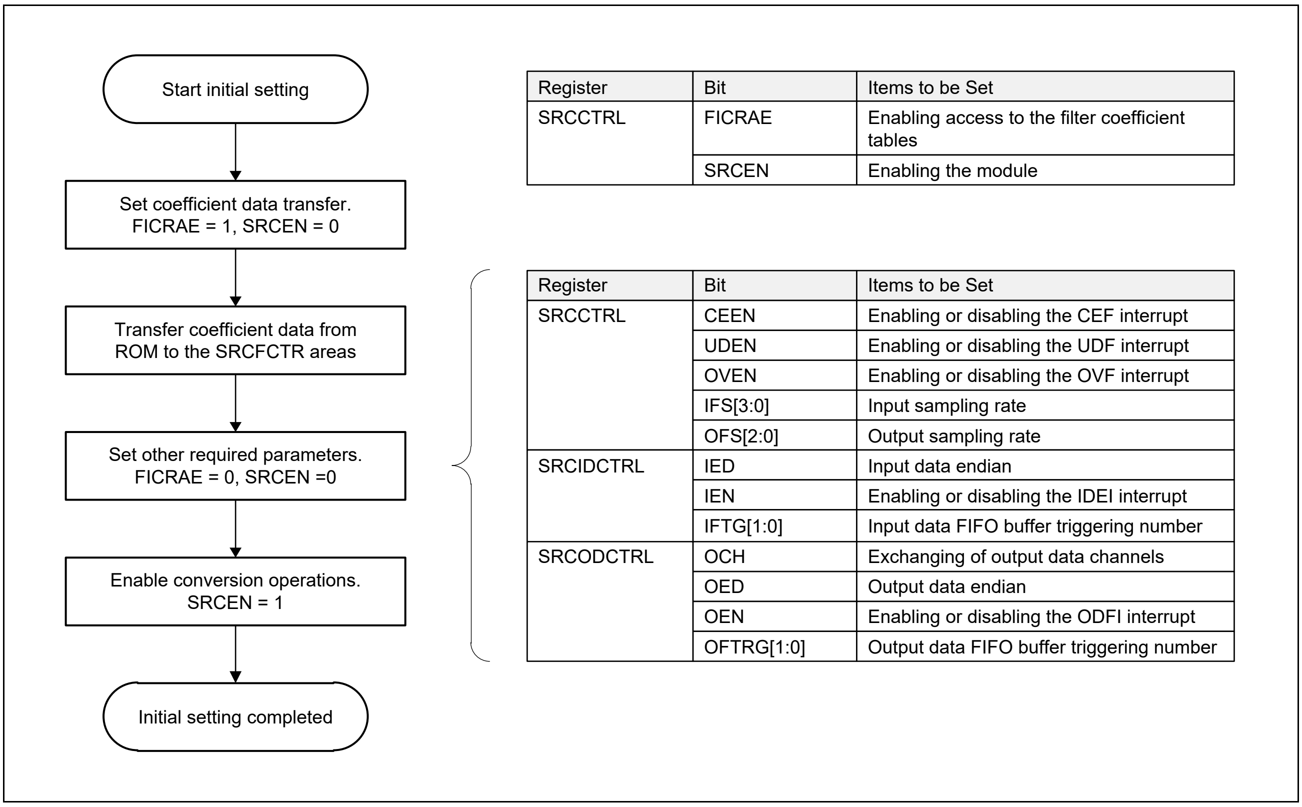
x: Undefined

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 0	SRCFCOE [21:0]	Undefined	R/W	Filter coefficient tables

29.4 Operation

29.4.1 Initial Setting

Figure 29.2 shows a sample flowchart for initial setting. After the reset release and before conversion operations in the SRC start, the filter coefficient data stored in a medium such as flash memory (ROM) must be transferred to the filter coefficient tables (SRCFCTRs). When the coefficient data have already been stored in the filter coefficient tables, the required parameters can be set immediately.



Register	Bit	Items to be Set
SRCCTRL	FICRAE	Enabling access to the filter coefficient tables
	SRCEN	Enabling the module

Register	Bit	Items to be Set
SRCCTRL	CEEN	Enabling or disabling the CEF interrupt
	UDEN	Enabling or disabling the UDF interrupt
	OVEN	Enabling or disabling the OVF interrupt
	IFS[3:0]	Input sampling rate
	OFS[2:0]	Output sampling rate
SRCIDCTRL	IED	Input data endian
	IEN	Enabling or disabling the IDEI interrupt
	IFTG[1:0]	Input data FIFO buffer triggering number
SRCODCTRL	OCH	Exchanging of output data channels
	OED	Output data endian
	OEN	Enabling or disabling the ODFI interrupt
	OFTRG[1:0]	Output data FIFO buffer triggering number

Figure 29.2 Sample Flowchart for Initial Setting

29.4.2 Data Input

Figure 29.3 is a sample flowchart for data input.

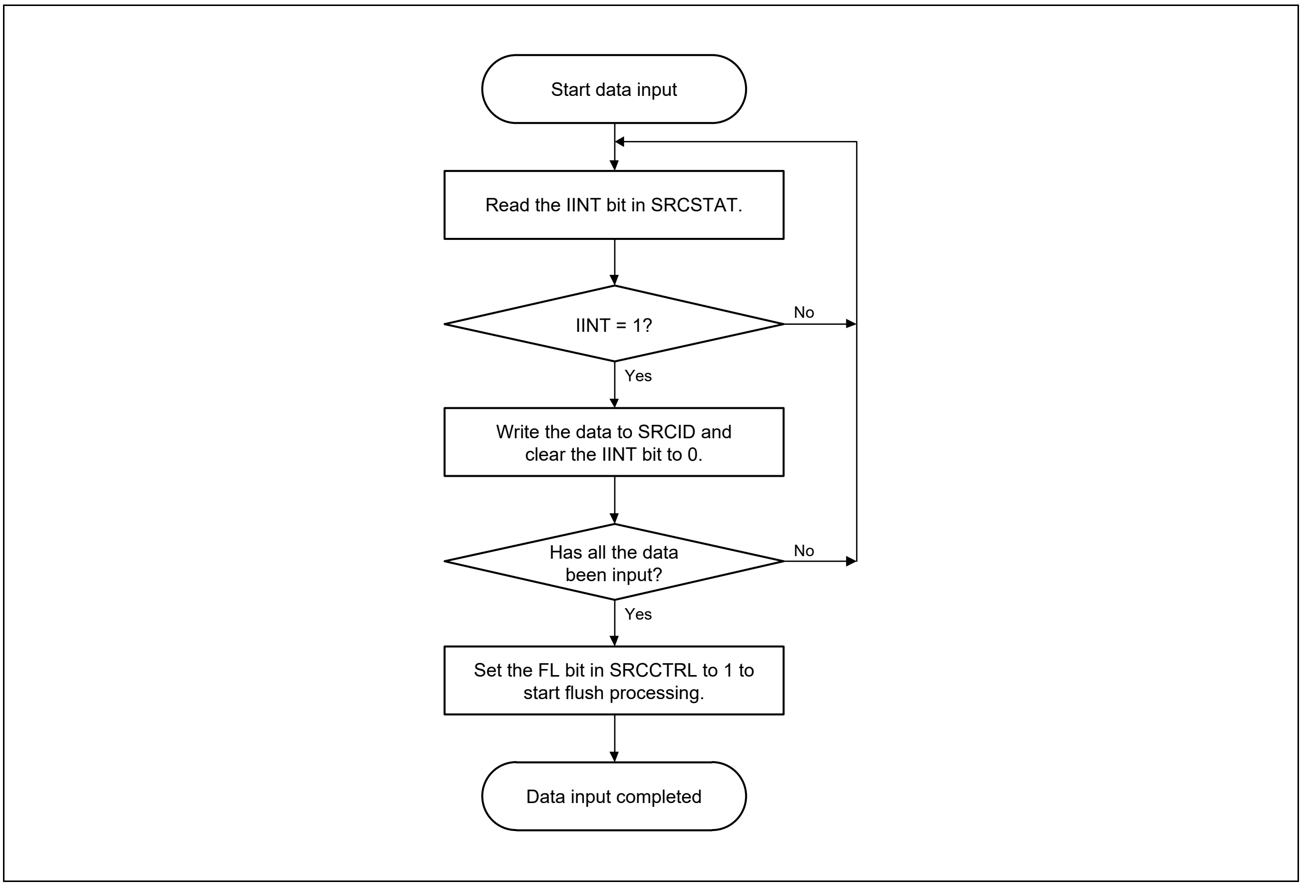


Figure 29.3 Sample Flowchart for Data Input

(1) When Interrupts are Issued to CPU

1. Set the IEN bit in SRCIDCTRL to 1.
2. When the IINT bit in SRCSTAT is set to 1, an IDEI interrupt request is issued. In the interrupt processing routine, read the IINT bit and confirm that it is 1, write data to SRCID. Then return from the interrupt processing routine.
3. Transfer the data by repeating step 2 as many times as required.
4. When all data have been input, write 1 to the FL bit in SRCCTRL.

(2) When Interrupts are Used to Activate Direct Memory Access Controller

1. Assign the IDEI interrupt request of this module to one channel of the direct memory access controller.
2. Set the IEN bit in SRCIDCTRL to 1.
3. When the IINT bit in SRCSTAT is set to 1, an IDEI interrupt request, that is, a DMA transfer request, is issued thus activating the direct memory access controller. When the direct memory access controller has written data to the SRCID thus resulting in the last data unit having been written, the IINT bit is cleared to 0.
4. Transfer the data by repeating step 3 as many times as required.
5. When all data have been input, write 1 to the FL bit in SRCCTRL.

NOTE

The number of stages of the input FIFO buffer is eight. Moreover, the possible number of data units that can be transferred in response to an IDEI interrupt request, that is, the number of empty FIFO buffer stages, depends on the setting of the IFTRG[1:0] bits in SRCCTRL. As the input FIFO buffer has no functionality to avoid or detect being overwritten, the data in it will be destroyed when it is overwritten. The number of contiguous data units to be transferred by DMA transfer, therefore, must be set in consideration of the setting of the IFTRG[1:0] bits.

29.4.3 Data Output

Figure 29.4 is a sample flowchart for data output.

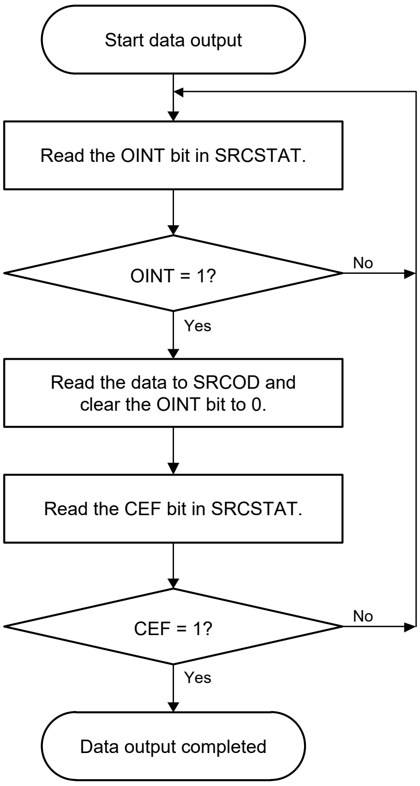


Figure 29.4 Sample Flowchart for Data Output

(1) When Interrupts are Issued to CPU

1. Set the OEN bit in SRCODCTRL to 1.
2. When the OINT bit in SRCSTAT is set to 1, an ODFI interrupt request is issued. In the interrupt processing routine, read the OINT bit and confirm that it is 1, read data from SRCOD. Then return from the interrupt processing routine.
3. Read the data by repeating step 2 as many times as required.
4. When all data have been read after flush processing starts, the CEF bit in SRCSTAT is set to 1. When the CEF bit is read as 1, the data output has been completed.

(2) When Interrupts are Used to Activate Direct Memory Access Controller

1. Assign the ODFI interrupt request of this module to one channel of the direct memory access controller.
2. Set the OEN bit in SRCODCTRL to 1.
3. When the OINT bit in SRCSTAT is set to 1, an ODFI interrupt request, that is, a DMA transfer request, is issued thus activating the direct memory access controller. When the direct memory access controller has read data from SRCOD thus resulting in the last data unit having been read, the OINT bit is cleared to 0.
4. Read the data by repeating step 3 as many times as required.
5. When all data have been read after flush processing starts, the CEF bit in SRCSTAT is set to 1. When the CEF bit is read as 1, the data output has been completed.

NOTES

1. The number of stages of the output FIFO buffer is sixteen. When the output FIFO buffer overflows, that is, when it is overwritten without data in it having been read, conversion processing is stopped. Though the output FIFO buffer can be read even when it has overflowed, resuming conversion processing requires additional processing that depends on the setting of the OVEN bit. For details, see the description of the OVEN bit in SRCCTR.
2. When the output FIFO buffer is read while the number of data units in it is zero, the value read is invalid. The number of contiguous data units to be read by DMA transfer must be set in consideration of the setting of the OFTRG[1:0] bits.

29.4.4 Interrupts

This module has five interrupt sources. **Table 29.7** summarizes the interrupts.

Table 29.7 Interrupt Requests and Generation Conditions

Interrupt Request	Abbreviation	Interrupt Condition	Direct Memory Access Controller Activation	Interrupt Request Signal
Input data FIFO buffer empty	IDEI	IINT = 1, IEN = 1, and SRCEN = 1	Possible	Edge
Output data FIFO buffer full	ODFI	OINT = 1, OEN = 1, and SRCEN = 1	Possible	Edge
Output data FIFO buffer overwrite	OVF	OVF = 1, OVEN = 1, and SRCEN = 1	Not possible	Level
Output data FIFO buffer underflow	UDF	UDF = 1, UDEN = 1, and SRCEN = 1	Not possible	Level
Conversion end	CEF	CEF = 1, CEEN = 1, and SRCEN = 1	Not possible	Level

When one of the interrupt conditions is met, the corresponding pin outputs an interrupt request signal that is detected as an edge or level. In response to the signal, the CPU executes the corresponding interrupt exception handling routine. The corresponding interrupt source flag, IINT, OINT, OVF, UDF, or CEF should be cleared from within the routine.

The IDEI and ODFI interrupts can activate the direct memory access controller (DMA) when the DMA is set to allow this. The IINT and OINT flags must not be cleared by writing from the CPU during DMA transfer. Clearing here refers to writing 0 to a bit after having read it as 1.

29.5 Usage Notes

29.5.1 Note on Accessing Registers

After the following write access to SRCCTRL, three cycles of the peripheral clock elapse before the corresponding bit in SRCSTAT is updated.

- Before the FLF bit in SRCSTAT is set after 1 is written to the FL bit in SRCCTRL
- Before each bit in SRCSTAT is initialized after 1 is written to the CL bit in SRCCTRL
- Before each bit in SRCSTAT is initialized after 1 is written to the SRCEN bit in SRCCTRL while the SRCEN bit is 0

As the CPU executes any subsequent instruction without waiting for the completion of the register writing, an instruction that immediately follows that used to write to SRCCTRL cannot accurately detect the updated state of SRCSTAT. To check the updated SRCSTAT state, dummy-read SRCCTRL or SRCSTAT after the instruction used to write to SRCCTRL.

29.5.2 Note on Flush Processing

When 1 is written to the FL bit in SRCCTRL, this module continues conversion processing by adding 0-data to the input data end point. Flush processing, therefore, should be performed when the audio data end point is input and there is no subsequent data.

To perform conversion again after flush processing, clear the internal working memory in either of the following ways.

- Write 1 to the CL bit in SRCCTRL.
- Write 0 and then 1 to the SRCEN bit in SRCCTRL.

29.5.3 Note on CPU Operations during Transfer by the DMAC

When the DMAC is used for the data transfer to the input and from the output data registers (SRCID and SRCOD), the IINT and OINT bits (bits 1 and 0) in the status register (SRCSTAT) must not be cleared by the CPU during transfer. Clearing here refers to writing 0 to a bit after having read it as 1.

29.5.4 Note on Access while the SRC is Operating

Access to the filter coefficient tables must not proceed while the SRC is operating, that is, while the setting of the SRCEN bit is 1.

29.5.5 Note on After rest

The coefficient data is not initialized by reset, so reloading of the coefficient data is not necessary.

30. Gigabit Ethernet Interface

Gigabit Ethernet Interface includes Ethernet controller (E-MAC) that conforms to the definition of the MAC (Media Access Control) layer for Ethernet in the IEEE 802.3 standard. When connected with a physical-layer LSI chip (PHY-LSI) that complies with the standard, the E-MAC is able to transmit and receive Ethernet (IEEE 802.3) frames. The E-MAC has a single MAC layer interface.

Internal TCP/IP Offload Engine (TOE) calculates Checksum by hardware and has functions filtering Ethernet Frames and automatically responding specific frames. Dedicated Direct memory access controller (DMAC) for transferring transmitted Ethernet frames to and received Ethernet frames from respective storage areas in the URAM at high speed.

30.1 Features

30.1.1 Specification

- 2 channels
- Supports transfer at 1000 Mbps and 100 Mbps, 10 Mbps
- Supports filtering of Ethernet frames
- Supports interface conforming to IEEE802.3 PHY RGMII (Reduced Gigabit Media Independent Interface)
- Supports interface conforming to IEEE802.3 PHYMII (Media Independent Interface)

Table 30.1 lists the features of the Gigabit Ethernet Interface block.

Table 30.1 Specifications (Functions) (1/2)

IP Name	1. Function
E-MAC	(1) Transmission and reception of Ethernet (IEEE 802.3) frames (2) Supports transfer at 10, 100 and 1000 Mbps in Full-duplex Mode, 10, 100 Mbps in Half-duplex Mode (3) Flow control conforming with the IEEE 802.3x standard (4) Supports MII (Media Independent Interface) and RGMII (Reduced Gigabit Media Independent Interface) (5) Transmission/Reception Low Power Idle Code
TOE	(1) Calculation of Checksum for following part of frames – IPv4 Header – IPv4 TCP/UDP/ICMP – IPv6 TCP/UDP/ICMP (2) Filtering of Ethernet Frames The Possible Configuration of Filter Condition are followings. – Ethernet Type – IP Protocol No. – UDP destination port No. – Destination MAC address (Unicast) – Destination MAC address (Broadcast) – Destination MAC address (Multicast) – ARP Request of Local Station – Neighbor Solicitation of Local Station – Except IPv6 Next Header (Analyzable) (3) Automatically Responding to ARP Request and Neighbor Solicitation

Table 30.1 Specifications (Functions) (2/2)

IP Name	1. Function
DMAC	(1) Supports AXI for DMA (128bit data width) (2) Supports APB for register access (32bit data width) (3) Direct Memory Access between URAM and TOE/E-MAC by Descriptor Method (4) Supported Maximum Frame Size are followings. – Transmission Frame is 1.5 KByte – Reception Frame is 8 Kbyte

30.1.2 Block Diagram

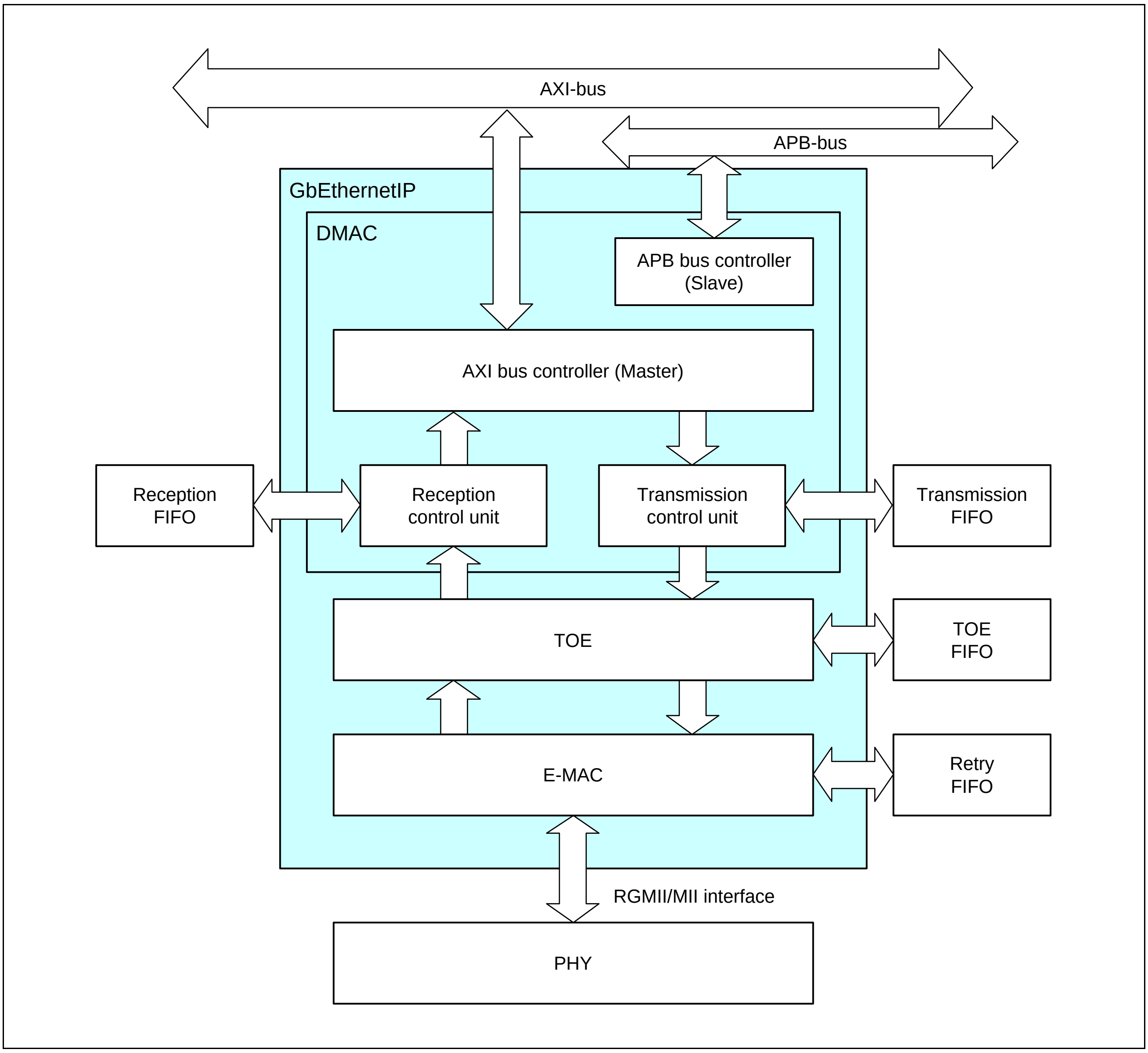


Figure 30.1 Block Diagram of Gigabit Ethernet Interface

30.2 External signal pins

Table 30.2 shows external signal pins for Gigabit Ethernet Interface.

Table 30.2 External pin Interface

Name	I/O	Description	Active Level	Initial value	Handling in not used
ET0_TXC/TX_CLK	I	Ch0 transmit clock signal	—	—	Pull-Down
ET0_TX_CTL/TX_EN	O	Ch0 Transmit control/enable signal	H	0b	Open
ET0_TXD[3-0]	O	Ch0 Transmit data signal	—	0b	Open
ET0_TX_ERR	O	Ch0 Transmit error signal	H	0b	Open
ET0_TX_COL	I	Ch0 Transmit collision signal	H	—	Pull-Down
ET0_TX_CRS	I	Ch0 Transmit carrier sense signal	H	—	Pull-Down
ET0_RXC/RX_CLK	I	Ch0 Receive clock signal	—	—	Pull-Down
ET0_RX_CTL/RX_DV	I	Ch0 Receive data Control/valid signal	H	—	Pull-Down
ET0_RXD[3-0]	I	Ch0 Receive data signal	—	—	Pull-Down
ET0_RX_ERR	I	Ch0 Receive error signal	H	—	Pull-Down
ET0_MDC	O	Ch0 Management information transfer clock signal	—	0b	Open
ET0_MDIO	I/O	Ch0 Management information transmit/receive data	—	Open	Pull-Down
ET0_LINKSTA	I	Ch0 PHY Link status signal	H	—	Pull-Down
ET1_TXC/TX_CLK	I	Ch1 transmit clock signal	—	—	Pull-Down
ET1_TX_CTL/TX_EN	O	Ch1 Transmit control/enable signal	H	0b	Open
ET1_TXD[3-0]	O	Ch1 Transmit data signal	—	0b	Open
ET1_TX_ERR	O	Ch1 Transmit error signal	H	0b	Open
ET1_TX_COL	I	Ch1 Transmit collision signal	H	—	Pull-Down
ET1_TX_CRS	I	Ch1 Transmit carrier sense signal	H	—	Pull-Down
ET1_RXC/RX_CLK	I	Ch1 Receive clock signal	—	—	Pull-Down
ET1_RX_CTL/RX_DV	I	Ch1 Receive data Control/valid signal	H	—	Pull-Down
ET1_RXD[3-0]	I	Ch1 Receive data signal	—	—	Pull-Down
ET1_RX_ERR	I	Ch1 Receive error signal	H	—	Pull-Down
ET1_MDC	O	Ch1 Management information transfer clock signal	—	0b	Open
ET1_MDIO	I/O	Ch1 Management information transmit/receive data	—	Open	Pull-Down
ET1_LINKSTA	I	Ch1 PHY Link status signal	H	—	Pull-Down

30.3 Register Configuration

30.3.1 Register Base Address

The base addresses of each channel are listed in the following table.

Table 30.3 Register Base Address

Register Name	Base Address Name	Base Address
ETH0	<ETH0_CA55_base>	H'0_11C2_0000
	<ETH0_CM33_NS_base>	H'41C2_0000
	<ETH0_CM33_S_base>	H'51C2_0000
ETH1	<ETH1_CA55_base>	H'0_11C3_0000
	<ETH1_CM33_NS_base>	H'41C3_0000
	<ETH1_CM33_S_base>	H'51C3_0000

30.3.2 IO Mode Registers

Refer to **Section 41, General Purpose Input Output Port (GPIO)** for Gigabit Ethernet Interface IO mode configuration.

30.3.3 DMAC Registers

Table 30.4 Configuration of DMAC-related Registers (From H'000 to H'4FF) (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
DMAC mode register	CCC	R/W	H'0000 0000	H'000	32
Descriptor base address table register	DBAT	R/W	H'0000 0000	H'004	32
Descriptor base address load request register	DLR	R/W	H'003F FFFF	H'008	32
DMAC status register	CSR	R	H'0000 0001	H'00C	32
Current descriptor address register q (q = 0,4)	CDARq	R	H'0000 0000	H'010 + q × 4	32
Error status register	ESR	R	H'0000 0000	H'088	32
Receive configuration register	RCR	R/W	H'6000 0000	H'090	32
Reception Truncation Configuration register	RTC	R/W	H'7FFC 7FFC	H'0B4	32
Transmit configuration register	TGC	R/W	H'0022 2200	H'300	32
Transmit configuration control register	TCCR	R/W	H'0000 0000	H'304	32
Transmit status register	TSR	R	H'0000 0000	H'308	32
MAC status FIFO Access register	MFA	R	H'0000 0000	H'30C	32
MAC status FIFO Access 2 register	MFA2	R	H'0000 0000	H'340	32
Descriptor interrupt control register	DIC	R/W	H'0000 0000	H'350	32
Descriptor interrupt status register	DIS	R/W	H'0000 0000	H'354	32
Error interrupt control register	EIC	R/W	H'0000 0000	H'358	32
Error interrupt status register	EIS	R/W	H'0000 0000	H'35C	32
Receive interrupt control register 0	RIC0	R/W	H'0000 0000	H'360	32
Receive interrupt status register 0	RIS0	R/W	H'0000 0000	H'364	32
Receive interrupt control register 1	RIC1	R/W	H'0000 0000	H'368	32

Receive interrupt status register 1	RIS1	R/W	H’0000 0000	H’36C	32
Receive interrupt control register 2	RIC2	R/W	H’0000 0000	H’370	32
Receive interrupt status register 2	RIS2	R/W	H’0000 0000	H’374	32

Table 30.4 Configuration of DMAC-related Registers (From H’000 to H’4FF) (2/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Transmit interrupt control register	TIC	R/W	H’0000 0000	H’378	32
Transmit interrupt status register	TIS	R/W	H’0000 0000	H’37C	32
Interrupt summary status register	ISS	R	H’0000 0000	H’380	32
Common Interrupt Enable register	CIE	R/W	H’0000 0000	H’384	32
Receive Interrupt Control register 3	RIC3	R/W	H’0000 0000	H’388	32
Receive Interrupt Status register 3	RIS3	R/W	H’0000 0000	H’38C	32
Descriptor Interrupt Enable register	DIE	R/W	H’0000 0000	H’450	32
Descriptor Interrupt Disable register	DID	R/W	H’0000 0000	H’454	32
Error Interrupt Enable register	EIE	R/W	H’0000 0000	H’458	32
Error Interrupt Disable register	EID	R/W	H’0000 0000	H’45C	32
Receive Interrupt Enable register 0	RIE0	R/W	H’0000 0000	H’460	32
Receive Interrupt Disable register 0	RID0	R/W	H’0000 0000	H’464	32
Receive Interrupt Enable register 1	RIE1	R/W	H’0000 0000	H’468	32
Receive Interrupt Disable register 1	RID1	R/W	H’0000 0000	H’46C	32
Receive Interrupt Enable register 2	RIE2	R/W	H’0000 0000	H’470	32
Receive Interrupt Disable register 2	RID2	R/W	H’0000 0000	H’474	32
Transmit Interrupt Enable register	TIE	R/W	H’0000 0000	H’478	32
Transmit Interrupt Disable register	TID	R/W	H’0000 0000	H’47C	32
Receive Interrupt Enable register 3	RIE3	R/W	H’0000 0000	H’488	32
Receive Interrupt Disable register 3	RID3	R/W	H’0000 0000	H’48C	32

30.3.4 E-MAC Registers

Table 30.5 Configuration of E-MAC-related Registers (From H’500 to H’7FF) (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
E-MAC operating mode register	CXR20	R/W	H’0000 0000	H’500	32
Maximum receive frame length register	CXR2A	R/W	H’0000 0000	H’508	32
Interrupt status register	CXR21	R/W	H’0000 000X	H’510	32
Enables/disables interrupts register	CXR22	R/W	H’0000 0000	H’518	32
PHY register access register	CXR23	R/W	H’0000 000X	H’520	32
Link status register	CXR2B	R	H’0000 000X	H’528	32
Polarity set register	CXR2C	R/W	H’0000 0000	H’52C	32
In-Band Status set register	CXR31	R/W	H’0000 0000	H’530	32
In-Band Status indicate register	CXR32	R	H’0000 000X	H’534	32
Mode indicate register	CXR33	R	H’0000 0000	H’538	32
PHY interface select register	CXR35	R/W	H’FFFF 0000	H’540	32
PHY interface indicate register	CXR36	R	H’0000 0000	H’544	32
PAUSE frame register 1	CXR71	R/W	H’0000 0000	H’554	32

PAUSE frame register 2	CXR72	R/W	H’0000 0000	H’558	32
PAUSE frame register 3	CXR8A	R/W	H’0000 0000	H’55C	32
PAUSE frame register 4	CXR80	R/W	H’0000 0000	H’560	32
PAUSE frame register 5	CXR81	R/W	H’0000 0000	H’564	32

Table 30.5 Configuration of E-MAC-related Registers (From H’500 to H’7FF) (2/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
PAUSE frame register 6	CXR82	R	H’0000 0000	H’568	32
E-MAC operating mode 2 register	CXR2D	R/W	H’0000 0000	H’5B0	32
Software LINK status register	CXR2G	R/W	H’0000 0000	H’5BC	32
Mac Address register 1	CXR24	R/W	H’0000 0000	H’5C0	32
Mac Address register 2	CXR25	R/W	H’0000 0000	H’5C8	32
TINT1 counter register	CXR40	R/W	H’0000 0000	H’700	32
TINT2 counter register	CXR41	R/W	H’0000 0000	H’708	32
TINT3 counter register	CXR42	R/W	H’0000 0000	H’710	32
RINT1 counter register	CXR50	R/W	H’0000 0000	H’740	32
RINT2 counter register	CXR51	R/W	H’0000 0000	H’748	32
RINT3 counter register	CXR52	R/W	H’0000 0000	H’750	32
RINT4 counter register	CXR53	R/W	H’0000 0000	H’758	32
RINT5 counter register	CXR54	R/W	H’0000 0000	H’760	32
RINT6 counter register	CXR55	R/W	H’0000 0000	H’768	32
RINT7 counter register	CXR56	R/W	H’0000 0000	H’770	32
RINT8 counter register	CXR57	R/W	H’0000 0000	H’778	32
MDIO status register	MDIOSTS	R	H’0000 0000	H’780	32
MDIO command register	MDIOCMD	R/W	H’0000 0000	H’784	32
MDIO address register	MDIOADR	R/W	H’0000 0000	H’788	32
MDIO data register	MDIODAT	R/W	H’0000 0000	H’78C	32
MDIO mode register	MDIOMOD	R/W	H’8000 00FF	H’790	32
Low Power Mode register 1	LPTXMOD1	R/W	H’0000 0000	H’7B0	32
Low Power Mode register 2	LPTXMOD2	R/W	H’0000 0000	H’7B4	32
RGMII Low Power parameter register 1	LPTXGTH1	R/W	H’0000 0000	H’7C0	32
RGMII Low Power parameter register 2	LPTXGTH2	R/W	H’0000 0000	H’7C4	32
RGMII Low Power parameter register 3	LPTXGTH3	R/W	H’0000 0000	H’7C8	32
RGMII Low Power parameter register 4	LPTXGTH4	R/W	H’0000 0000	H’7CC	32
MII Low Power parameter register 1	LPTXMTH1	R/W	H’0000 0000	H’7D0	32
MII Low Power parameter register 2	LPTXMTH2	R/W	H’0000 0000	H’7D4	32
MII Low Power parameter register 3	LPTXMTH3	R/W	H’0000 0000	H’7D8	32
MII Low Power parameter register 4	LPTXMTH4	R/W	H’0000 0000	H’7DC	32

30.3.5 TOE Registers

Table 30.6 Configuration of TOE-related Registers (From H’800 to H’9FF) (1/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Checksum operating mode register	CSR0	R/W	H’0000 0000	H’800	32
Tx Frame Checksum Enable register	CSR1	R/W	H’0000 0000	H’804	32
Rx Frame Checksum Enable register	CSR2	R/W	H’0000 0000	H’808	32
Tx Extended Header Number register	CSR3	R/W	H’0000 0008	H’80C	32
Rx Extended Header Number register	CSR4	R/W	H’0000 0008	H’810	32
Unsupported Tx Frame Counter register	CSR20	R/W	H’0000 0000	H’820	32
Unsupported Rx Frame Counter register	CSR30	R/W	H’0000 0000	H’824	32
Rx IPv4 Header checksum Error Counter register	CSR31	R/W	H’0000 0000	H’828	32
Rx TCP/UDP/ICMP checksum Error Counter register	CSR32	R/W	H’0000 0000	H’82C	32
Filter Operation/Auto Response Enable register	CSFR00	R/W	H’1020 0000	H’840	32
Local IPv4 address register	CSFR01	R/W	H’0000 0000	H’844	32
Local IPv6 address register i (i=0 to 3)	CSFR02_i	R/W	H’0000 0000	H’848 + 4 × i	32
Upper Local MAC address register	CSFR03_U	R/W	H’0000 0000	H’858	32
Lower Local MAC address register	CSFR03_L	R/W	H’0000 0000	H’85C	32
IPv6 no_next_header Protocol Number register	CSFR04	R/W	H’0000 003B	H’860	32
Ethernet Type Condition register i (i=0 to 3)	CSFR10_i	R/W	H’0000 0000	H’870 + 4 × i	32
Ethernet Type Condition Enable register	CSFR10	R/W	H’0000 0000	H’880	32
Protocol Condition register i (i=0 to 3)	CSFR11_i	R/W	H’0000 0000	H’884 + 4 × i	32
Protocol Condition Enable register	CSFR11	R/W	H’0000 0000	H’894	32
UDP Por Condition register i (i=0 to 11)	CSFR12_i	R/W	H’0000 0000	H’898 + 4 × i	32
UDP Port Condition Enable register	CSFR12	R/W	H’0000 0000	H’8C8	32
Upper MAC DA unicast address Condition register	CSFR13_U	R/W	H’0000 0000	H’8CC	32
Lower MAC DA unicast address Condition register	CSFR13_L	R/W	H’0000 0000	H’8D0	32
Upper MAC DA broadcast address Condition register	CSFR14_U	R/W	H’0000 FFFF	H’8D4	32
Lower MAC DA broadcast address Condition register	CSFR14_L	R/W	H’FFFF FFFF	H’8D8	32
Upper MAC DA multicast address Condition register i (i=0 to 19)	CSFR15_U_i	R/W	H’0000 0100	H’8DC + 8 × i	32
Lower MAC DA multicast address Condition register i (i=0 to 19)	CSFR15_L_i	R/W	H’5E00 0000	H’8E0 + 8 × i	32
MAC DA Condition Enable register	CSFR15	R/W	H’0000 0000	H’97C	32
IPv6 analysis Protocol Condition register 0	CSFR16_0	R/W	H’2911 0600	H’980	32
IPv6 analysis Protocol Condition register 1	CSFR16_1	R/W	H’3A33 2C2B	H’984	32
IPv6 analysis Protocol Condition register 2	CSFR16_2	R/W	H’0000 003C	H’988	32
IPv6 analysis Protocol Condition Enable register	CSFR16	R/W	H’0000 01FF	H’98C	32
Wake Up Interrupt Status register	CSFR20	R/W	H’0000 0000	H’9A0	32
Wake Up Interrupt Mask register	CSFR21	R/W	H’101F FFFF	H’9A4	32

Table 30.6 Configuration of TOE-related Registers (From H’800 to H’9FF) (2/2)

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Auto Response Configuration register1	CSFR30	R/W	H’0001 7B35	H’9B0	32
Auto Response Configuration register2	CSFR31	R/W	H’0000 0000	H’9B4	32
ARPREQ/Neighbor Solicitations Receive Interrupt Status register	CSFR40	R/W	H’0000 0000	H’9C0	32
ARPREQ/Neighbor Solicitations Receive Interrupt Mask register	CSFR41	R/W	H’0000 0007	H’9C4	32
Rx ARPREQ/Neighbor Solicitation Upper MAC SA register	CSFR42_U	R	H’0000 0000	H’9C8	32
Rx ARPREQ/Neighbor Solicitation Lower MAC SA register	CSFR42_L	R	H’0000 0000	H’9CC	32
Rx ARPREQ/Neighbor Solicitation IP SA register i (i=0 to 3)	CSFR43_i	R	H’0000 0000	H’9D0 + 4 × i	32

30.4 Register Descriptions

30.4.1 DMAC Registers

30.4.1.1 DMAC Mode Register (CCC)

The CCC register specifies the operating mode of the DMAC.

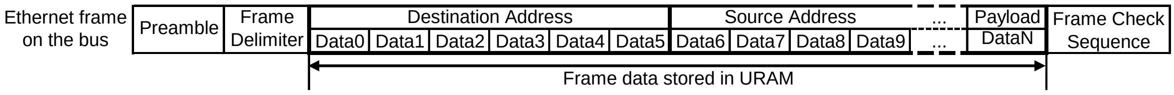
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	FCE	—	—	—	—	BOC	—	ERCS	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RDFD	DTSR	—	—	—	—	—	—	OPC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25	FCE	0b	R/W	Flow Control Enable 0b Flow control disabled 1b Flow control enabled This bit enables the flow control support of E-MAC. When flow control is enabled, the E-MAC gets informed about the Reception FIFO level (Reception FIFO fill level reached RCR.RFCL).
24 to 21	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	BOC	0b	R/W	Byte Order Configuration 0b First Ethernet byte in URAM [7:0] This bit defines the byte ordering of frame data in URAM in relation to the byte order in the Ethernet frame. The CPU can only write 0b to this bit.*[1]
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	ERCS	0b	R/W	Enable Reference 250MHz Clock Stop function 0b Disable reference 250 MHz clock(clk_miitx_gtx_refclk) stop function. 1b Enable reference 250 MHz clock(clk_miitx_gtx_refclk) stop function. This bit enables the stop function of reference clock of Ethernet. When this bit is '1', Gigabit Ethernet Interface asserts Output Signal “refclk_stop_en” during it does not need reference clock. *Note*: If PHY does not support stopping TXC of RGMII Interface, set “0” to this bit. Because TXC is generated from clk_miitx_gtx_refclk. [Changing condition] This bit is set to 0b when RESET mode is entered.
17 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	RDFD	0b	R/W	Reception Descriptor Fetch Disable 0b Reception Descriptor Fetch Enable 1b Reception Descriptor Fetch Disable This register is used to stop the fetch of reception descriptor. By fetch stops, the state of Reception Descriptor Controller of DMAC will stop at IDLE. CSR.RDFDM indicates the current fetch status of reception descriptor. The CPU should not change any configuration or mode until CSR.RDFDM is updated once the Reception Descriptor Fetch Disable request is given. The CPU should not cancel the Reception Descriptor Fetch Disable request (set this bit to 0b) before CSR.RDFDM is read as 1b.
8	DTSR	0b	R/W	Data Transfer Suspend Request 0b Normal operation 1b Request suspension of data transfer This bit requests to suspend further Gigabit Ethernet Interface accesses to URAM. A currently on-going access is completed. CSR.DTS indicates the current suspend status. The CPU should not change any configuration or mode until CSR.DTS is updated once data transfer suspend request is given. The CPU should not cancel the data transfer suspend request (set this bit to 0b) before CSR.DTS is read as 1b.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	OPC	00b	R/W	OPeration Command 00b Enter RESET mode 01b Enter CONFIG mode 10b Enter OPERATION mode 11b Reserved These bits define the operating mode DMAC should enter. Refer to **Section 4.1** for details about the operating modes. The CPU can write to these bits in any operating mode. [Changing condition] These bits are set to 00b when RESET mode is entered.

Note 1. Following illustrates a received Ethernet frame and the part of the frame stored in URAM.



When CCC.BOC is 0b, an Ethernet frame is represented in URAM as:

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPTR+0	Data3								Data2								Data1								Data0							
DPTR+4	Data7								Data6								Data5								Data4							
DPTR+8	Data11								Data10								Data9								Data8							

30.4.1.2 Descriptor Base Address Table Register (DBAT)

The DBAT register specifies the base address of the descriptor table in the URAM.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TA[31:0]	H'0000_0000	R/W	Table Address Address of descriptor base address table in URAM These bits define the address of the first entry of the descriptor base address table. Refer to **Section 4.2.3** about function and structure of the descriptor base address table. The CPU can only write to these bits if CSR.OPS is CONFIG and OPERATION. The CPU should only write values which are a multiple of 4 to these bits.

30.4.1.3 Descriptor Base Address Load Request Register (DLR)

The DLR register is used to issue a request to load the values from the count descriptor address register q (CDARq) for each queue to the descriptor base address table register (DBAT).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	LBA4	—	—	—	LBA0
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 5	—	All 1	R	Reserved Whenever it is read, all bits 1 are read. The written value will be ignored.
4	LBA4	1b	R/W	Load Base Address 4(Rx0) 0b No pending load request 1b Pending load request This bit requests to set the current descriptor address of queue q (CDARq.CDA) to an entry of the descriptor base address table (DBAT.TA + 8*4). By setting this bit to 1b CPU requests a load of the current descriptor address of queue q. If there is an on-going transfer for queue q, the load is processed after the whole frame has been transferred. Note: In case of a transmission queue, the load function is not influencing the transmit start request (TCCR.TSRQ); so, fetching continues at base address. To prevent this behavior, it is recommended to request the load of transmission queues only when TCCR.TSRQ is 0b. The CPU can only write to this bit if CSR.OPS is OPERATION. The CPU can only write 1b to this bit. [Changing condition] This bit is set to 1b when leaving OPERATION mode. This bit is set to 0b when there is no on-going transfer (related to old chain) for queue q. Note for verification: In Rx, if the entirety or the final part of a frame is stored into a FEMPTY_ND descriptor, then there is no actual frame transfer to URAM but still LBA reload is processed only when the transfer from RxRAM to BMI is completed for that frame.
3 to 1	—	All 1	R	Reserved Whenever it is read, all bits 1 are read. The written value will be ignored.
0	LBA0	1b	R/W	Load Base Address 0(Tx0) 0b No pending load request 1b Pending load request This bit requests to set the current descriptor address of queue q (CDARq.CDA) to an entry of the descriptor base address table (DBAT.TA + 8*0). For the other description, refer to LBA4 bit of this register.

30.4.1.4 DMAC Status Register (CSR)

The CSR register is used to indicate the operating mode in which the DMAC is running and the individual communications states.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	TDUO	RPO	—	—	—	TPO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RCSI	RDFDM	DTS	—	—	—	—	OPS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	TDUO	0b	R	Transmission Descriptor Update On-going 0b No pending descriptor update 1b Pending descriptor update in URAM This bit indicates that there is pending descriptor update for a transmit queue. When this bit is 1b, there are descriptors of already or currently processed transmit storage elements not updated in URAM. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when all transmit queue related storage elements processed. This bit is set to 1b when descriptor related status update procedure starts (descriptor update and SFR flagging).
20	RPO	0b	R	Reception Process On-going 0b No pending reception 1b Pending received frames in Reception FIFO This bit indicates that there is pending receive data for a receive queues. When this bit is 1b, there are frames in Reception FIFO that are not completely stored in URAM. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when a frame is completely stored in URAM, the descriptor update has been completed, and there are no pending frames in Reception FIFO for storage to URAM. This bit is set to 1b when a frame received by E-MAC is completely stored in Reception FIFO.
19 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TPO	0b	R	Transmission Process On-going 0b No pending transmission 1b Pending frames for transmission This bit indicates that there is pending or on-going transmission process for transmit queue. When this bit is 1b, a frame fetch from the queue in URAM has been started or completed but this frame has not been completely transmitted by E-MAC Descriptor write back is not included in this bit.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
10	RCSI	0b	R	Reference 250 MHz Clock is being Stopped Internally 0b Reference 250 MHz Clock is being not stopped internally 1b Reference 250 MHz Clock is being stopped internally This bit indicates that Gigabit Ethernet Interface has stopped Reference 250 MHz Clock (clk_miitx_gtx_refclk) internally or not. When this bit is 1b, Internal Gated Clock Cell for Reference 250 MHz Clock is closed. During that time, this clock is not provided to internal logics. [Changing condition] This bit is set to 0b when the condition of input signal “refclk_stop” changes to 0b. This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when CCC.ERCS is 0b. This bit is set to 1b when the condition of output signal “refclk_stop_en” changes to 1b.
9	RDFDM	0b	R	Reception Descriptor Fetch Disable Monitor 0b Reception Descriptor Fetch will happen 1b Reception Descriptor Fetch is stopped This bit is for monitoring. It can be used in order to confirm a state of Reception Descriptor Controller of DMAC. If this bit is '1', it means the fetch of Reception Descriptor from URAM by DMAC is completely stopped. [Changing condition] This bit is set to 0b when CCC.RDFD is 0b. This bit is set to 0b when RESET mode is entered. This bit is set to 1b when CCC.RDFD is 1b and fetch of Reception Descriptor from URAM is completely stopped.
8	DTS	0b	R	Data Transfer Suspended 0b Access to URAM will happen 1b Access to URAM is suspended This bit indicates if DMAC is allowed to access URAM. [Changing condition] This bit is set to 0b when CCC.DTSR is 0b. This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when CCC.DTSR is 1b and there is no on-going URAM access.
7 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	OPS	0001b	R	OPerating mode Status 0001b RESET 0010b CONFIG 0100b OPERATION others reserved These bits represent the current operating mode of DMAC. [Changing condition] These bits are updated when an operating mode changes is processed.

30.4.1.5 Current Descriptor Address Register q (CDARq) (q = 0,4)

The CDARq register indicates the current descriptor address.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CDA	All 0	R	Current Descriptor Address Current descriptor address of transmission / receive queue q These bits indicate the address of the descriptor which is currently being processed by queue q or which will be processed when queue q gets active. For q=0 the addresses refer to the transmit queues. From q=4 the addresses refer to the receive queues. *Note*: The current descriptor address cannot be used for HW/SW synchronization. CDARq.CDA may one or more descriptors ahead from information available in URAM. [Changing condition] These bits are set to 0 when leaving OPERATION mode. These bits are set to DBAT.TA + 8*q when entering OPERATION mode. These bits are set to DBAT.TA + 8*q when the load base address request of queue (DLR.LBA) has been processed. These bits are set to DESCR.DPTR when a link descriptor (LINK, LINKFIX) has been processed. These bits are incremented by the size of the descriptor control structure (8 for normal) when a frame data descriptor has been processed.

30.4.1.6 Error Status Register (ESR)

Error Status Register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	EIL	ET[3:0]				—	—	—	EQN[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	EIL	0b	R	Error Information Lost 0b No loss of error information 1b Lost of error information detected This bit indicates that error information detected by Gigabit Ethernet Interface is lost because the previous reported error has not been processed by CPU. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when CPU writes 0b to EIS.QEF. This bit is set to 1b when the set condition of EIS.QEF is fulfilled while EIS.QEF is 1b.
11 to 8	ET	0000b	R	Error Type 0000b (0) Read descriptor from URAM 0001b (1) Write descriptor to URAM 0010b (2) Interpret read descriptor 0011b (3) Transmission FIFO is corrupted 0100b (4) Read data from URAM 0101b (5) Write data to URAM 0110b (6) Reading data from Reception FIFO 0111b (7) Reception FIFO is corrupted 1000b (8) Frame size error during reception detected 1001b (9) Reserved 1010b (10) Transmission FIFO Data access error others reserved These bits indicate details about the transfer stage which was handled when Gigabit Ethernet Interface has detected an error. When the fault is related to the read descriptor (ESR.ET=0 or 2), CPU needs to correct the faulty descriptor before the related queue can continue processing. In this case the queue halts at the faulty descriptor and CDARq.CDA (with q=ESR.EQN) identifies faulty descriptor. All other errors are transient in nature and may be corrected by continuation of HW or SW operation; so there is no strong demand on CPU interaction. The CPU should only evaluate these bits when EIS.QEF is 1b. [Changing condition] These bits are updated when the set condition of EIS.QEF is fulfilled and EIS.QEF is 0b.
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	EQN	00000b	R	Number of faulty queue These bits indicate the queue number which was being handled when Gigabit Ethernet Interface has detected an error. A fault reported for ESR.EQN= 0 is related to transmit queue. From ESR.EQN=4 the fault is related to receive queue. The CPU should only evaluate these bits when EIS.QEF is 1b. The CPU should not evaluate these bits when ESR.ET is 0011b, 0111b. [Changing condition] These bits are updated when the set condition of EIS.QEF is fulfilled and EIS.QEF is 0b.

30.4.1.7 Receive Configuration Register (RCR)

The RCR register is used to make settings related to reception for the DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RFCL[14:0]														
Initial Value	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EFFS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 16	RFCL[14:0]	H’6000	R/W	Reception FIFO Critical Level Number of bytes stored in Reception FIFO before critical level is reached (RFCL[1:0] are fixed to 0). These bits define a fill level of the Reception FIFO in bytes to generate FIFO critical level reached warning notification (RIS1.RFWF). The CPU can only write to these bits if CSR.OPS is CONFIG. The CPU should only write values which are a multiple of 4 to these bits.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	EFFS	0b	R/W	Enable Faulty Frame Storage 0b Storage of faulty frames disabled 1b Storage of faulty frames enabled (for debug) This bit enables the storage of received frames classified as faulty by the TOE or E-MAC. Faulty received frames will always be stored in receive queue. *Note*: DESCR.MSC informs about the reception fault detected by the TOE or E-MAC. The CPU can only write to this bit if CSR.OPS is CONFIG. *Note*: This bit is used only for debug. Do not set RCR.EFFS=1 during normal operation.

30.4.1.8 Reception Truncation Configuration Register I (RTC)

Reception Truncation Configuration register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	MFL[14:0]														
Initial Value	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 18	—	All 1	R	Reserved Whenever it is read, all bits 1 are read. The written value will be ignored.
17 to 15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14 to 0	MFL	H’7FFC	R/W	Maximum Frame Length These bits define the maximum frame length (64 to 8k bytes) stored to URAM. When a frame received by E-MAC is truncated, CPU gets informed by EIS.QEF and frame size error (ESR.ET is 1000b). Additionally, the DESCR.TI in the updated descriptor is set to 1b. The CPU can only write to these bits if CSR.OPS is CONFIG. The CPU should only write values which are a multiple of 4 to these bits. The CPU should not write values less than 64 to these bits.

30.4.1.9 Transmit Configuration Register (TGC)

The TGC register is used to make settings related to transmission for the DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TBD[1:0]		—	—	—	—	—	—	—	—
Initial Value	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
20 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
16 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	—	1b	R	Reserved Whenever it is read, 1 is read. The written value will be ignored.
12 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9 to 8	TBD	10b	R/W	Transmission FIFO Depth This bit defines the depth of the Transmission FIFO used for transmit queue. Gigabit Ethernet Interface fetches up to TGC.TBD frames from URAM to Transmission FIFO for transmit queue. The CPU can only write “10b” to this bit. Frame transmission by E-MAC can start for transmission queue when at least one frame from transmit queue is available in Transmission FIFO. The CPU can only write to these bits if CSR.OPS is CONFIG.
7 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.10 Transmit Configuration Control Register (TCCR)

The TCCR register controls transmission by the DMAC and is used to make related settings.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	MFR2	MFEN2	MFR	MFEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	MFR2	0b	R/W	MAC status FIFO2 Release 0b No requests to MAC status FIFO2 1b Release oldest entry of MAC status FIFO2 This bit releases the oldest entry of the MAC status FIFO2. When 1b is written to this bit, Gigabit Ethernet Interface gets informed that CPU has processed the oldest MAC status FIFO2 entry as visible in MFA2 register. The oldest entry is removed from MAC status FIFO2. This bit is always read as 0b. *Note*: Do not use the function of this bit in the Duplex Mode (CXR20.DPM=1).
18	MFEN2	0b	R/W	MAC status FIFO2 Enable 0b Disabled 1b Enabled This bit enables the storage of transmission status information (the E-MAC flags for frame transmission) in the MAC status FIFO2. When this bit is set to 0b all pending FIFO entries are removed. *Note*: Do not use the function of this bit in the Duplex Mode (CXR20.DPM=1).
17	MFR	0b	R/W	MAC status FIFO Release 0b No request to MAC status FIFO 1b Release oldest entry of MAC status FIFO This bit releases the oldest entry of the MAC status FIFO. When 1b is written to this bit, DMAC gets informed that CPU has processed the oldest MAC status FIFO entry as visible in MFA register. The oldest entry is removed from MAC status FIFO. This bit is always read as 0b.
16	MFEN	0b	R/W	MAC status FIFO Enable 0b Disabled 1b Enabled This bit enables the storage of transmission status information (the TOE flags for frame transmission) in the MAC status FIFO. Additionally, CPU can control the status storage for each frame provided for transmission using DESCR.MSR. When this bit is set to 0b all pending FIFO entries are removed.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	TSRQ	0b	R/W	Transmission Start Request Queue 0b Transmission queue empty or stopped 1b Pending frames in queue to fetch This bit defines that frames, provided by CPU in transmit queue in URAM should be transmitted. When this bit is 1b, there are pending frames in transmit queue which have not been fetched to the Transmission FIFO. *Note*: The transmission start request cannot be used for HW/SW synchronization. TCCR.TSRQ may be cleared before frame data is completely fetched from URAM. *Note*: The frame transmission by E-MAC is decoupled from the fetching to the Transmission FIFO. Transmission scheduling for a queue depends on the transmission priority. CSR.TPO gives an indication about pending transmissions. The CPU can only write to this bit if CSR.OPS is OPERATION. The CPU can only write 1b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when CPU writes 1b to it. This bit is set to 0b when DMAC has processed an EEMPTY, FEMPTY or LEMPTY descriptor (no data available). This bit is set to 0b when DMAC has processed an EOS descriptor (end of set). This bit is set to 0b when DMAC has processed a faulty descriptor. This bit is set to 0b when CCC.OPC is 01b and there is no on-going fetch transfer for queue.

30.4.1.11 Transmit Status Register (TSR)

The TSR register indicates the state of transmission by the DMAC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	MFFL2[4:0]					—	—	—	MFFL[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28 to 24	MFFL2	0000b	R	MAC status FIFO2 Fill Level Number of entries stored in the MAC status FIFO2 These bits indicate the number of entries available in the MAC status FIFO2. If this value is 0, the FIFO2 is empty. If this value is 16, the FIFO2 is full. The values 17 to 31 are reserved. [Changing condition] These bits are set to 0 when leaving OPERATION mode. These bits are set to 0 when TCCR.MFEN2 is 0b. This value is incremented when TCCR.MFEN2 is 1b, TSR.MFFL2 is not 16, and E-MAC detects an error during transmission of frame. This value is decremented when 1b is written to TCCR.MFR2 and TSR.MFFL2 is not 0.
23 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20 to 16	MFFL	00000b	R	MAC status FIFO Fill Level Number of entries stored in the MAC status FIFO These bits indicate the number of entries available in the MAC status FIFO. If this value is 0, the FIFO is empty. If this value is 16, the FIFO is full. The values 17 to 31 are reserved. [Changing condition] These bits are set to 0 when leaving OPERATION mode. These bits are set to 0 when TCCR.MFEN is 0b. This value is incremented when frame with DESCR.MSR has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is not 16. This value is decremented when 1b is written to TCCR.MFR and TSR.MFFL is not 0.
15 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.12 MAC status FIFO Access Register (MFA)

MAC status FIFO Access register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MST[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MSV[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25to 16	MST	H'00	R	MAC Status Tag Tag number from descriptor identifying the frame MAC status relation These bits represent the DESCR.TAG bits from the descriptor defining the data for frame transmission. The tag is used to get the relation between the frame inside a transmit queue and the TOE status available in the FIFO (MFA.MSV). The CPU should not read these bits when TSR.MFFL is 0. [Changing condition] These bits are updated when the first entry is stored in FIFO (TSR.MFFL changes from 0 to 1). These bits are updated when the oldest entry is released (writing 1b to TCCR.MFR). These bits are updated when the FIFO is overwritten (a frame with DESCR.MSR in descriptor has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is 16). Note for verification: These bits are set to 0 when leaving OPERATION mode.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9to 0	MSV	H'00	R	MAC status value These bits represent the TOE status bits as stored in the oldest MAC status FIFO entry. MSV [9] Unsupported transmission frame MSV [8:0] Reserved The CPU should not read these bits when TSR.MFFL is 0. [Changing condition] These bits are updated when the first entry is stored in FIFO (TSR.MFFL changes from 0 to 1). These bits are updated when the oldest entry is released (writing 1b to TCCR.MFR). These bits are updated when the FIFO overwritten (a frame with DESCR.MSR in descriptor has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is 16). Note for verification: These bits are set to 0 when leaving OPERATION mode.

30.4.1.13 MAC status FIFO Access 2 Register (MFA2)

MAC status FIFO Access 2 register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MSV2[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8 to 0	MSV2	H’00	R	MAC Status Value2 These bits represent the MAC status bits as stored in the oldest MAC status FIFO2 entry. MSV [8] abort information MSV [7:3] Reserved MSV [2] Carrier lost during frame transmission MSV [1] Delay collision MSV [0] Transmission time out The CPU should not read these bits when TSR.MFFL2 is 0. [Changing condition] These bits are updated when the first entry is stored in FIFO2 (TSR.MFFL2 changes from 0 to 1). These bits are updated when the oldest entry is released (writing 1b to TCCR.MFR2). These bits are updated when the FIFO2 is overwritten when TCCR.MFEN2 is 1b, TSR.MFFL2 is 16, and E-MAC detects an error during transmission of frame.

30.4.1.14 Descriptor Interrupt Status Register (DIS)

The DIS register indicates the state of descriptor interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPF15	DPF14	DPF13	DPF12	DPF11	DPF10	DPF9	DPF8	DPF7	DPF6	DPF5	DPF4	DPF3	DPF2	DPF1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPFi	All 0	R/W	Descriptor Processed Flag i (i = 1 to 15) 0b No interrupt pending 1b Descriptor interrupt pending This bit indicates that a descriptor in a reception or transmit queue has been processed where DESCR.DIE is i. *Note*: When DESCR.DIE is 0000b, no descriptor interrupt is generated. *Note*: When DESCR.DIE is 0001b, in addition a queue specific descriptor interrupt is generated. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a descriptor has been processed where DESCR.DIE is i.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.15 Descriptor Interrupt Control Register (DIC)

The DIC register is used to control descriptor interrupts 1 to 15.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPE15	DPE14	DPE13	DPE12	DPE11	DPE10	DPE9	DPE8	DPE7	DPE6	DPE5	DPE4	DPE3	DPE2	DPE1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPEi	All 0	R/W	Descriptor Processed interrupt Enable i (i = 1 to 15) 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when DIS.DPFi is 1b. [Changing condition] This bit is set to 0b when writing 1b to DID.DPDi. This bit is set to 1b when writing 1b to DIE.DPSi.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.16 Descriptor Interrupt Enable Register (DIE)

The DIE register set to 1 each bit in descriptor interrupt control register (DIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPS15	DPS14	DPS13	DPS12	DPS11	DPS10	DPS9	DPS8	DPS7	DPS6	DPS5	DPS4	DPS3	DPS2	DPS1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPSi	All 0	R/W	Descriptor Processed interrupt Set i (i = 1 to 15) 0b No change of DIC.DPEi 1b Set DIC.DPEi to 1b This bit supports interrupt enable. It controls set of DIC.DPEi. This bit is always read as 0b.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.17 Descriptor Interrupt Disable Register (DID)

The DID register set to 0 each bit in descriptor interrupt control register (DIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPD15	DPD14	DPD13	DPD12	DPD11	DPD10	DPD9	DPD8	DPD7	DPD6	DPD5	DPD4	DPD3	DPD2	DPD1	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 1	DPDi	All 0	R/W	Descriptor Processed interrupt Disable i (i = 1 to 15) 0b No change of DIC.DPEi 1b Set DIC.DPEi to 0b This bit supports interrupt enable. It controls set of DIC.DPEi. This bit is always read as 0b.
0	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.18 Error Interrupt Status Register (EIS)

The EIS register indicates the states of DMAC-related error interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFF2	—	MFFF	—	—	—	—	—	—	QEF	MTEF	MREF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	QFS	0b	R	Queue Full Summary 0b No interrupt pending 1b Queue full interrupt pending Summary bit to indicate that at least one queue is full. [Changing condition] This bit is set when any matching pair of RIC2.QFE enable and RIS2.QFF flag are both 1b; or when both the RIS2.RFFF flag and RIC2.RFFE enable are 1b.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFF2	0b	R/W	MAC status FIFO2 Full Flag 0b No interrupt pending 1b MAC status FIFO2 full, oldest MAC status lost MAC status FIFO2 Full Flag This bit indicates that a transmission MAC status is overwritten because the MAC status FIFO2 is full (overwrite condition). The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when TCCR.MFEN2 is 1b, TSR.MFFL2 is 16, and E-MAC detects an error during transmission of frame.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFF	0b	R/W	MAC status FIFO Full Flag 0b No interrupt pending 1b MAC status FIFO full, oldest MAC status lost This bit indicates that a transmission MAC status is overwritten because the MAC status FIFO is full (overwrite condition). The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when frame with DESCR.MSR has been transmitted to TOE, TCCR.MFEN is 1b and TSR.MFFL is 16.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
2	QEF	0b	R/W	Queue Error Flag 0b No interrupt pending 1b Interrupt pending This bit indicates that an error has been detected while processing receive or transmit queue. Detail about the detected error is indicated by ESR. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when an error condition is detected.
1	MTEF	0b	R/W	MAC Transmission Error Flag 0b No interrupt pending 1b MAC has reported an error during transmission This bit indicates that the E-MAC has detected a fault during transmission. For detail, the E-MAC registers have to be checked. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when E-MAC detects an error during frame transmission.
0	MREF	0b	R/W	MAC Reception Error Flag 0b No interrupt pending 1b MAC has reported an error during reception This bit indicates that the E-MAC has detected a fault during reception. For detail, the E-MAC registers have to be checked. *Note*: When the storage of faulty received frames (RCR.EFFS) is enabled, the E-MAC error code is stored in the descriptor (DESCR.MSC). By evaluating this information CPU can identify corrupted frames in URAM. The CPU can only write 0b to this bit.

30.4.1.19 Error Interrupt Control Register (EIC)

The EIC register controls the DMAC-related error interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFE2	—	MFFE	—	—	—	—	—	—	QEE	MTEE	MREE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFE2	0b	R/W	MAC status FIFO2 Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MFFF2 is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MFFD2. This bit is set to 1b when writing 1b to EIE.MFFS2.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFE	0b	R/W	MAC status FIFO Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MFFF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MFFD. This bit is set to 1b when writing 1b to EIE.MFFS.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	QEE	0b	R/W	Queue Error interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.QEF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.QED. This bit is set to 1b when writing 1b to EIE.QES.
1	MTEE	0b	R/W	MAC Transmission Error interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MTEF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MTED. This bit is set to 1b when writing 1b to EIE.MTES.

Bit	Bit Name	Initial Value	R/W	Description
0	MREE	0b	R/W	MAC Reception Error interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when EIS.MREF is 1b. [Changing condition] This bit is set to 0b when writing 1b to EID.MRED. This bit is set to 1b when writing 1b to EIE.MRES.

30.4.1.20 Error Interrupt Enable Register (EIE)

The EIE register set to 1 each bit in error interrupt control register (EIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFS2	—	MFFS	—	—	—	—	—	—	QES	MTES	MRES
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFS2	0b	R/W	MAC status FIFO2 Full interrupt Set 0b No change of EIC.MFFE2 1b Set EIC.MFFE2 to 1b This bit supports interrupt enable. It controls set of EIC.MFFE2. This bit is always read as 0b.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFS	0b	R/W	MAC status FIFO Full interrupt Set 0b No change of EIC.MFFE 1b Set EIC.MFFE to 1b This bit supports interrupt enable. It controls set of EIC.MFFE. This bit is always read as 0b.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	QES	0b	R/W	Queue Error interrupt Set 0b No change of EIC.QEE 1b Set EIC.QEE to 1b This bit supports interrupt enable. It controls set of EIC.QEE. This bit is always read as 0b.
1	MTES	0b	R/W	MAC Transmission Error interrupt Set 0b No change of EIC.MTEE 1b Set EIC.MTEE to 1b This bit supports interrupt enable. It controls set of EIC.MTEE. This bit is always read as 0b.
0	MRES	0b	R/W	MAC Reception Error interrupt Set 0b No change of EIC.MREE 1b Set EIC.MREE to 1b This bit supports interrupt enable. It controls set of EIC.MREE. This bit is always read as 0b.

30.4.1.21 Error Interrupt Disable Register (EID)

The EID register set to 0 each bit in error interrupt control register (EIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MFFD2	—	MFFD	—	—	—	—	—	—	QED	MTED	MRED
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	MFFD2	0b	R/W	MAC status FIFO2 Full interrupt Disable 0b No change of EIC.MFFE 1b Set EIC.MFFE to 0b This bit supports interrupt enable. It controls set of EIC.MFFE2. This bit is always read as 0b.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	MFFD	0b	R/W	MAC status FIFO Full interrupt Disable 0b No change of EIC.MFFE 1b Set EIC.MFFE to 0b This bit supports interrupt enable. It controls set of EIC.MFFE. This bit is always read as 0b.
8 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	QED	0b	R/W	Queue Error interrupt Disable 0b No change of EIC.QEE 1b Set EIC.QEE to 0b This bit supports interrupt enable. It controls set of EIC.QEE. This bit is always read as 0b.
1	MTED	0b	R/W	MAC Transmission Error interrupt Disable 0b No change of EIC.MTEE 1b Set EIC.MTEE to 0b This bit supports interrupt enable. It controls set of EIC.MTEE. This bit is always read as 0b.
0	MRED	0b	R/W	MAC Reception Error interrupt Disable 0b No change of EIC.MREE 1b Set EIC.MREE to 0b This bit supports interrupt enable. It controls set of EIC.MREE. This bit is always read as 0b.

30.4.1.22 Receive Interrupt Status Register 0 (RIS0)

The RIS0 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRF	0b	R/W	Frame Received Flag 0b No interrupt pending 1b New frame data is available in URAM This bit indicates that in receive queue r a frame is stored, and data is available to be processed by CPU. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a frame is stored in receive queue.

30.4.1.23 Receive Interrupt Control Register 0 (RIC0)

The RIC0 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRE	0b	R/W	Frame Received interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS0.FRF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID0.FRD. This bit is set to 1b when writing 1b to RIE0.FRS.

30.4.1.24 Receive Interrupt Enable Register 0 (RIE0)

The RIE0 register set to 1 each bit in receive interrupt control register (RIC0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRS	0b	R/W	Frame Received interrupt Set r 0b No change of RIC0.FRE 1b Set RIC0.FRE to 1b This bit supports interrupt enable. It controls set of RIC0.FRE. This bit is always read as 0b.

30.4.1.25 Receive Interrupt Disable Register 0 (RID0)

The RID0 register set to 0 each bit in receive interrupt control register (RIC0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRD	0b	R/W	Frame Received interrupt Disable 0b No change of RIC0.FRE 1b Set RIC0.FRE to 0b This bit supports interrupt enable. It controls set of RIC0.FRE. This bit is always read as 0b.

30.4.1.26 Receive Interrupt Status Register 1 (RIS1)

The RIS1 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWF	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWF	0b	R/W	Reception FIFO Warning Flag 0b No interrupt pending 1b Reception FIFO warning level reached This bit indicates that the Reception FIFO has reached the configured critical warning level. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a complete frame is received in the Reception FIFO and the fill level goes above the configured critical warning level during reception of this frame (RCR.RFCL).
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.27 Receive Interrupt Control Register 1 (RIC1)

The RIC1 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWE	0b	R/W	Reception FIFO Warning interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS1.RFWF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID1.RFWD. This bit is set to 1b when writing 1b to RIE1.RFWS.
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.28 Receive Interrupt Enable Register 1 (RIE1)

The RIE register set to 1 each bit in error receive control register (RIC1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWS	0b	R/W	Reception FIFO Warning interrupt Set 0b No change of RIC1.RFWE 1b Set RIC1.RFWE to 1b This bit supports interrupt enable. It controls set of RIC1.RFWE. This bit is always read as 0b.
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.29 Receive Interrupt Disable Register 1 (RID1)

The RID1 register set to 1 each bit in receive interrupt control register (RIC1).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFWD	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFWD	0b	R/W	Reception FIFO Warning interrupt Disable 0b No change of RIC1.RFWE 1b Set RIC1.RFWE to 0b This bit supports interrupt enable. It controls set of RIC1.RFWE. This bit is always read as 0b.
30 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

30.4.1.30 Receive Interrupt Status Register 2 (RIS2)

The RIS2 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFF	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFF	0b	R/W	Reception FIFO Full Flag 0b No interrupt pending 1b Reception FIFO full, received frame data lost This bit indicates that a frame was received by E-MAC which could not be completely stored in the Reception FIFO. The incomplete frame is lost. There is no information available which receive queue would handle this frame. Frames which are marked by E-MAC as faulty after complete reception can also trigger an overrun. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when frame data provided by E-MAC cannot be stored in the Reception FIFO.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFF	0b	R/W	Queue Full Flag 0b No interrupt pending 1b Queue full, received frame data lost This bit indicates that there was no sufficient space in receive queue to store completely a received frame. A receive queue is treated as full when no empty descriptor (read descriptor with DESCR.DT = FEMPTY or FEMPTY_ND) is available. *Note*: If there is no empty descriptor left in queue during storage of a split frame, an incomplete frame will be stored in the queue. Such incomplete frame can be identified based on a descriptor sequence error (no FEND). *Note*: Reading descriptor may be one or more storage elements ahead from actual storage in URAM. CDARq.CDA provides information where queue has been stopped. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when there is pending receive data in Reception FIFO for receive queue and there is no space left for storage. This bit is set to 1b when a SW defined stop point (EOS descriptor) reached within a split frame.

30.4.1.31 Receive Interrupt Control Register 2 (RIC2)

The RIC2 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFE	0b	R/W	Reception FIFO Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS2.RFFF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID2.RFFD. This bit is set to 1b when writing 1b to RIE2.RFFS.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFE	0b	R/W	Queue Full interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS2.QFF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID2.QFD. This bit is set to 1b when writing 1b to RIE2.QFS.

30.4.1.32 Receive Interrupt Enable Register 2 (RIE2)

The RIE2 register set to 1 each bit in receive interrupt control register (RIC2).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFS	0b	R/W	Reception FIFO Full interrupt Set 0b No change of RIC2.RFFE 1b Set RIC2.RFFE to 1b This bit supports interrupt enable. It controls set of RIC2.RFFE. This bit is always read as 0b.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFS	0b	R/W	Queue Full interrupt Set 0b No change of RIC2.QFE 1b Set RIC2.QFE to 1b This bit supports interrupt enable. It controls set of RIC2.QFE. This bit is always read as 0b.

30.4.1.33 Receive Interrupt Disable Register 2 (RID2)

The RID2 register set to 1 each bit in receive interrupt control register (RIC2).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFFD	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	QFD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RFFD	0b	R/W	Reception FIFO Full interrupt Disable 0b No change of RIC2.RFFE 1b Set RIC2.RFFE to 0b This bit supports interrupt enable. It controls set of RIC2.RFFE. This bit is always read as 0b.
30 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	QFD	0b	R/W	Queue Full interrupt Disable 0b No change of RIC2.QFE 1b Set RIC2.QFE to 0b This bit supports interrupt enable. It controls set of RIC2.QFE. This bit is always read as 0b.

30.4.1.34 Receive Interrupt Status Register 3 (RIS3)

The RIS3 register indicates the states of the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPF	0b	R/W	Receive Descriptor Processed Flag 0b No interrupt pending 1b Descriptor interrupt pending This bit indicates that a descriptor in reception queue has been processed where DESCR.DIE is 0001b. *Note*: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a descriptor in reception queue has been processed where DESCR.DIE is 1.

30.4.1.35 Receive Interrupt Control Register 3 (RIC3)

The RIC3 register controls the DMAC receive interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPE	0b	R/W	Receive Descriptor Processed interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when RIS3.RDPF is 1b. [Changing condition] This bit is set to 0b when writing 1b to RID3.RDPD. This bit is set to 1b when writing 1b to RIE3.RDPS.

30.4.1.36 Receive Interrupt Enable Register 3 (RIE3)

The RIE3 register set to 1 each bit in receive interrupt control register (RIC3).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPS	0b	R/W	Receive Descriptor Processed interrupt Set 0b No change of RIC3.RDPE 1b Set RIC3.RDPE to 1b This bit supports interrupt enable. It controls set of RIC3.RDPE. This bit is always read as 0b.

30.4.1.37 Receive Interrupt Disable Register 3 (RID3)

The RID3 register set to 0 each bit in receive interrupt control register (RIC3).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDPD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RDPD	0b	R/W	Receive Descriptor Processed interrupt Disable 0b No change of RIC3.RDPE 1b Set RIC3.RDPE to 0b This bit supports interrupt enable. It controls set of RIC3.RDPE. This bit is always read as 0b.

30.4.1.38 Transmit Interrupt Status Register (TIS)

The TIS register indicates the states of the DMAC transmit interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRF	MFWF2	MFUF2	MFWF	MFUF	—	—	—	—	—	—	—	—	—	FTF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPF	0b	R/W	Transmit Descriptor Processed Flag 0b No interrupt pending 1b Descriptor interrupt pending This bit indicates that a descriptor in transmit queue has been processed where DESCR.DIE is 0001b. *Note*: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a descriptor in transmit queue has been processed where DESCR.DIE is 1.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRF	0b	R/W	Reference Clock Stop Req Flag 0b No interrupt pending 1b Reference Clock Stop Request occurred This bit indicates that Gigabit Ethernet Interface expects to stop “clk_miitx_gtx_refclk”. Whenever this flag is asserted, CPU has to stop ETH_PLL (PLL to generate “clk_miitx_gtx_refclk”). To stop ETH_PLL, refer to the document of System Controller. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when CCC.ERCS is 0b. This bit is set to 1b when Gigabit Ethernet Interface has expected to stop “clk_miitx_gtx_refclk”.

Bit	Bit Name	Initial Value	R/W	Description
13	MFWF2	0b	R/W	MAC status FIFO2 Warning Flag 0b No interrupt pending 1b Tx Status FIFO2 warning level has been reached This bit indicates that the warning level of the MAC status FIFO2 (12 out of 16 entries) has been reached. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN2 is 0b. This bit is set to 0b when writing 1b to TCCR.MFR2. This bit is set to 1b when TCCR.MFEN2 is 1b, the MAC Status FIFO2 contains already 11 entries (TSR.MFFL2 is 11), and E-MAC detects an error during transmission of frame.
12	MFUF2	0b	R/W	MAC status FIFO2 Updated Flag 0b No interrupt pending 1b Tx Status FIFO2 has been updated This bit indicates that the MAC status FIFO2 has been updated when the E-MAC detects an error during transmission of frame. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN2 is 0b. This bit is set to 0b when writing 1b to TCCR.MFR2. This bit is set to 1b when TCCR.MFEN2 is 1b and E-MAC detects an error during transmission of frame.
11	MFWF	0b	R/W	MAC status FIFO Warning Flag 0b No interrupt pending 1b Tx Status FIFO warning level has been reached This bit indicates that the warning level of the MAC status FIFO (12 out of 16 entries) has been reached. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN is 0b. This bit is set to 0b when writing 1b to TCCR.MFR. This bit is set to 1b when frame with DESCR.MSR has been transmitted to TOE, TCCR.MFEN is 1b and the MAC Status FIFO contains already 11 entries (TSR.MFFL is 11).
10	MFUF	0b	R/W	MAC status FIFO Updated Flag 0b No interrupt pending 1b Tx Status FIFO has been updated This bit indicates that the MAC status FIFO has been updated after the TOE has transmitted a frame. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 0b when TCCR.MFEN is 0b. This bit is set to 0b when writing 1b to TCCR.MFR. This bit is set to 1b when frame with DESCR.MSR has been transmitted to TOE and TCCR.MFEN is 1b.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	FTF	0b	R/W	Frame Transmitted Flag 0b No interrupt pending 1b Frame transmitted by E-MAC This bit indicates that from transmit queue a frame is transmitted from DMAC. The CPU can only write 0b to this bit. [Changing condition] This bit is set to 0b when leaving OPERATION mode. This bit is set to 1b when a frame from transmit queue has been transmitted from DMAC.

30.4.1.39 Transmit Interrupt Control Register (TIC)

The TIC register controls the DMAC transmit interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRE	MFWE2	MFUE2	MFWE	MFUE	—	—	—	—	—	—	—	—	—	FTE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPE	0b	R/W	Transmit Descriptor Processed interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.TDPF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.TDPD. This bit is set to 1b when writing 1b to TIE.TDPS.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRE	0b	R/W	Reference Clock Stop Req interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.RCSRF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.RCSRD. This bit is set to 1b when writing 1b to TIE.RCSRS.
13	MFWE2	0b	R/W	MAC status FIFO2 Warning interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFWF2 is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFWD2. This bit is set to 1b when writing 1b to TIE.MFWS2.
12	MFUE2	0b	R/W	MAC status FIFO2 Updated interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFUF2 is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFUD2. This bit is set to 1b when writing 1b to TIE.MFUS2.

Bit	Bit Name	Initial Value	R/W	Description
11	MFWE	0b	R/W	MAC status FIFO Warning interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFWF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFWD. This bit is set to 1b when writing 1b to TIE.MFWS.
10	MFUE	0b	R/W	MAC status FIFO Updated interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.MFUF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.MFUD. This bit is set to 1b when writing 1b to TIE.MFUS.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FTE	0b	R/W	Frame Transmitted interrupt Enable 0b Disabled 1b Enabled While this bit is 1b an interrupt will be generated when TIS.FTF is 1b. [Changing condition] This bit is set to 0b when writing 1b to TID.FTD. This bit is set to 1b when writing 1b to TIE.FTS.

30.4.1.40 Transmit Interrupt Enable Register (TIE)

The TIE register set to 1 each bit in transmit interrupt control register (TIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRS	MFWS2	MFUS2	MFWS	MFUS	—	—	—	—	—	—	—	—	—	FTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPS	0b	R/W	Transmit Descriptor Processed interrupt Set 0b No change of TIC.TDPE 1b Set TIC.TDPE to 1b This bit supports interrupt enable. It controls set of TIC.TDPE. This bit is always read as 0b.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRS	0b	R/W	Reference Clock Stop Req interrupt Set 0b No change of TIC.RCSRE 1b Set TIC.RCSRE to 1b This bit supports interrupt enable. It controls set of TIC.RCSRE. This bit is always read as 0b.
13	MFWS2	0b	R/W	MAC status FIFO2 Warning interrupt Set 0b No change of TIC.MFWE2 1b Set TIC.MFWE2 to 1b This bit supports interrupt enable. It controls set of TIC.MFWE2. This bit is always read as 0b.
12	MFUS2	0b	R/W	MAC status FIFO2 Updated interrupt Set 0b No change of TIC.MFUE2 1b Set TIC.MFUE2 to 1b This bit supports interrupt enable. It controls set of TIC.MFUE2. This bit is always read as 0b.
11	MFWS	0b	R/W	MAC status FIFO Warning interrupt Set 0b No change of TIC.MFWE 1b Set TIC.MFWE to 1b This bit supports interrupt enable. It controls set of TIC.MFWE. This bit is always read as 0b.
10	MFUS	0b	R/W	MAC status FIFO Updated interrupt Set 0b No change of TIC.MFUE 1b Set TIC.MFUE to 1b This bit supports interrupt enable. It controls set of TIC.MFUE. This bit is always read as 0b.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	FTS	0b	R/W	Frame Transmitted interrupt Set 0b No change of TIC.FTE 1b Set TIC.FTE to 1b This bit supports interrupt enable. It controls set of TIC.FTE. This bit is always read as 0b.

30.4.1.41 Transmit Interrupt Disable Register (TID)

The TID register set to 0 each bit in transmit interrupt control register (TIC).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDPD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCSRD	MFWD2	MFUD2	MFWD	MFUD	—	—	—	—	—	—	—	—	—	FTD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	TDPD	0b	R/W	Transmit Descriptor Processed interrupt Disable 0b No change of TIC.TDPE 1b Set TIC.TDPE to 0b This bit supports interrupt enable. It controls set of TIC.TDPE. This bit is always read as 0b.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	RCSRD	0b	R/W	Reference Clock Stop Req interrupt Disable 0b No change of TIC.RCSRE 1b Set TIC.RCSRE to 0b This bit supports interrupt enable. It controls set of TIC.RSCRE. This bit is always read as 0b.
13	MFWD2	0b	R/W	MAC status FIFO2 Warning interrupt Disable 0b No change of TIC.MFWE2 1b Set TIC.MFWE2 to 0b This bit supports interrupt enable. It controls set of TIC.MFWE2. This bit is always read as 0b.
12	MFUD2	0b	R/W	MAC status FIFO2 Updated interrupt Disable 0b No change of TIC.MFUE2 1b Set TIC.MFUE2 to 0b This bit supports interrupt enable. It controls set of TIC.MFUE2. This bit is always read as 0b.
11	MFWD	0b	R/W	MAC status FIFO Warning interrupt Disable 0b No change of TIC.MFWE 1b Set TIC.MFWE to 0b This bit supports interrupt enable. It controls set of TIC.MFWE. This bit is always read as 0b.
10	MFUD	0b	R/W	MAC status FIFO Updated interrupt Disable 0b No change of TIC.MFUE 1b Set TIC.MFUE to 0b This bit supports interrupt enable. It controls set of TIC.MFUE. This bit is always read as 0b.
9 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	FTD	0b	R/W	Frame Transmitted interrupt Disable 0b No change of TIC.FTE 1b Set TIC.FTE to 0b This bit supports interrupt enable. It controls set of TIC.FTE. This bit is always read as 0b.

30.4.1.42 Common Interrupt Enable register (CIE)

The CIE register is used to control interrupts line.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CTIE	—	—	—	—	—	—	—	CRIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	CTIE	0b	R/W	Common Transmit Interrupt Enable 0b Disabled 1b Enabled This bit controls transmit related interrupt outputs of Gigabit Ethernet Interface. It has no influence on internal flagging in Gigabit Ethernet Interface SFR (e.g. ISS).
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CRIE	0b	R/W	Common Receive Interrupt Enable 0b Disabled 1b Enabled This bit controls receive related interrupt outputs of Gigabit Ethernet Interface. It has no influence on internal flagging in Gigabit Ethernet Interface SFR (e.g. ISS).

30.4.1.43 Interrupt Summary Status Register (ISS)

The ISS register gives a summary of the states of DMAC-related interrupts.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DPM15	DPM14	DPM13	DPM12	DPM11	DPM10	DPM9	DPM8	DPM7	DPM6	DPM5	DPM4	DPM3	DPM2	DPM1	RCSRM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MFWM2	MFUM2	—	RFWM	MFWM	MFUM	—	—	MM	EM	—	—	—	FTM	—	FRM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	DPMi	All 0	R	Descriptor Processed Mirror i 0b No interrupt pending 1b Descriptor interrupt pending Descriptor Processed Mirror i (i = 1 to 15) [Changing condition] This bit is set when any matching pair of DIC.DPEi enable and DIS.DPFi flag are both 1b.
16	RCSRM	0b	R	Reference Clock Stop Req Mirror 0b No interrupt pending 1b Reference Clock Stop Req interrupt pending This bit indicates that TIS.RCSRF is 1b and TIC.RCSRE is 1b. [Changing condition] This bit is updated when TIS.RCSRF or TIC.RCSRE changes.
15	MFWM2	0b	R	MAC status FIFO2 Warning Mirror 0b No interrupt pending 1b MAC status FIFO2 warning interrupt pending This bit indicates that TIS.MFWF2 is 1b and TIC.MFWE2 is 1b. [Changing condition] This bit is updated when TIS.MFWF2 or TIC.MFWE2 changes.
14	MFUM2	0b	R	MAC status FIFO2 Updated Mirror 0b No interrupt pending 1b MAC status FIFO2 updated interrupt pending This bit indicates that TIS.MFUF2 is 1b and TIC.MFUE2 is 1b. [Changing condition] This bit is updated when TIS.MFUF2 or TIC.MFUE2 changes.
13	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	RFWM	0b	R	Reception FIFO Warning Summary 0b No interrupt pending 1b Reception FIFO warning interrupt pending This bit indicates that RIS1.RFWF is 1b and RIC1.RFWE is 1b. [Changing condition] This bit is updated when RIS1.RFWF or RIC1.RFWE changes.

Bit	Bit Name	Initial Value	R/W	Description
11	MFWM	0b	R	MAC status FIFO Warning Mirror 0b No interrupt pending 1b MAC status FIFO warning interrupt pending This bit indicates that TIS.MFWF is 1b and TIC.MFWE is 1b. [Changing condition] This bit is updated when TIS.MFWF or TIC.MFWE changes.
10	MFUM	0b	R	MAC status FIFO Updated Mirror 0b No interrupt pending 1b MAC status FIFO updated interrupt pending This bit indicates that TIS.MFUF is 1b and TIC.MFUE is 1b. [Changing condition] This bit is updated when TIS.MFUF or TIC.MFUE changes.
9 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	MM	0b	R	MAC Mirror 0b No interrupt pending 1b MAC interrupt pending This bit indicates that there is a pending interrupt request issued by MAC. *Note*: The MAC is able to assert interrupt also in CONFIG mode.
6	EM	0b	R	Error Mirror 0b No interrupt pending 1b Error interrupt pending This bit indicates that at least one enabled flag in EIS is 1b or EIS.QFS is 1b. [Changing condition] This bit is updated when EIS or EIC changes.
5 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	FTM	0b	R	Frame Transmitted Mirror 0b No interrupt pending 1b Frame transmitted interrupt pending [Changing condition] This bit is set when any matching pair of TIC.FTE enable and TIS.FTF flag are both 1b. This bit is set when any matching pair of TIC.TDPE enable and TIS.TDPF flag are both 1b.
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FRM	0b	R	Frame Received Mirror 0b No interrupt pending 1b Frame received interrupt pending [Changing condition] This bit is set when any matching pair of RIC0.FRE enable and RIS0.FRF flag are both 1b. This bit is set when any matching pair of RIC3.RDPE enable and RIS3.RDPF flag are both 1b.

30.4.2 E-MAC Registers

30.4.2.1 E-MAC operating mode register 1 (CXR20)

CXR20 sets the E-MAC operating mode.

Rewriting the bits in this register is prohibited when the transmission function is enabled (TPE=1) and reception function is enabled (RPE=1).

Write restrictions:

> Rewrite all the bits except for TPE and RPE when transmission function is disabled (TPE=0) and reception function is disabled (RPE=0).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TRCCM	RCPT	TCPT	RCSC	CXSER	DPAD	RZPF	TZPF	PFR	RXF	TXF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CER	—	—	—	—	—	RPE	TPE	LPM	—	—	DPM	PRM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	TRCCM	0b	R/W	TINT/RINT Counter Clear Mode Specifies the method of clearing TINT source counter (CXR40-43) and RINT source counter (CXR50-57). 1: Clears the register to 0 when reading the register. 0: Clears the register to 0 when writing to the register. (clears any write data to 0 at write)
25	RCPT	0b	R/W	Reception CRC Pass Through 1: CRC of received frame is transferred to TOE. RCSC (auto calculation of checksum of received frame data part) function is disabled at this time. 0: CRC of received frame is not transferred to TOE.
24	TCPT	0b	R/W	Transmit CRC Pass Through 1: CRC of transmit frame is not calculated automatically (hardware calculation). It is necessary to input the data with CRC added in AT port. 0: CRC of transmit frame is calculated automatically (hardware calculation). It is necessary to input the data without CRC added in AT port.
23	RCSC	0b	R/W	Reception Check Sum Calculation 1: Checksum of received frame data part is automatically calculated (hardware calculation). RCPT must be set to 0 at this time. 0: Checksum of received frame data part is not calculated automatically (hardware calculation).
22	CXSER	0b	R/W	Carrier extension Short Error Function for debug. It should always be set to 0.

Bit	Bit Name	Initial Value	R/W	Description
21	DPAD	0b	R/W	Data Padding 1: Less than 60-byte data is transmitted without padding. 0: Less than 60-byte data is padded and transmitted as 60-byte data.
20	RZPF	0b	R/W	Reception Zero PAUSE Frame 1: Enables to receive PAUSE frame with TIME parameter value being 0. 0: Disables receiving PAUSE frame with TIME parameter value being 0.
19	TZPF	0b	R/W	Transmit Zero PAUSE Frame 1: PAUSE frame of 0 Time parameter value is transmitted at asserting “Reception FIFO Critical Level Notification” from DMAC 0: PAUSE frame of 0 Time parameter value is not transmitted at asserting “Reception FIFO Critical Level Notification” from DMAC
18	PFR	0b	R/W	PAUSE Frame Receive 1: PAUSE frame is transferred to TOE. 0: PAUSE frame is not transferred to TOE.
17	RXF	0b	R/W	Receive Flow control mode 1: Receive flow control function (PAUSE frame reception) is enabled. 0: Receive flow control function (PAUSE frame reception) is disabled.
16	TXF	0b	R/W	Transmit Flow control mode 1: Transmit flow control function (PAUSE frame transmission) is enabled. 0: Transmit flow control function (PAUSE frame transmission) is disabled.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	CER	0b	R/W	CRC Error Used for debugging. It should always be set to 0.
11 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
8 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	RPE	0b	R/W	Receive Port Enable 1: Receive function of E-MAC is enabled. 0: Receive function of E-MAC is disabled. If 1 is changed to 0 while a frame is being received, the frame is completely received.
5	TPE	0b	R/W	Transmit Port Enable 1: Transmission function of E-MAC is enabled. 0: Transmission function of E-MAC is disabled. If 1 is changed to 0 while a frame is being transmitted, the frame is completely transmitted.
4	LPM	0b	R/W	Low Power Mode 1: E-MAC operates as Low Power Mode. 0: E-MAC operates as Normal Mode. (When LPM bit is set to “1”, DPM bit should be also set to “1”)
3 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DPM	0b	R/W	Duplex Mode 1: Operates in full duplex mode. 0: Operates in half duplex mode.

Bit	Bit Name	Initial Value	R/W	Description
0	PRM	0b	R/W	Promiscuous Mode 1: All the frames except for PAUSE frame are received. Self-addressed unicast, different address unicast, multicast, and broadcast frames are all transferred to TOE. PAUSE frame reception is controlled by PFR bit. 0: Self-addressed unicast, multicast, and broadcast frames are received, then transferred to TOE.

30.4.2.2 Maximum receive frame length register (CXR2A)

CXR2A sets the maximum receive frame length.

Write restrictions:

If CXR20_bit6 is set to 1, it is not allowed to rewrite to this register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LEN_LMT[17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LEN_LMT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17 to 0	LEN_LMT	H’000	R/W	frame LENgth upper LiMiT value Sets the maximum receive frame length (Including CRC). If the receive frame length exceeds the set value, frame length error (RINT4) is asserted. < Set value > < Check value > H’00000 to H’005EE 1518 Byte H’005EF 1519 Byte H’005F0 1520 Byte … H’007FF 2047 Byte H’00800 2048 Byte … H’01000 4096 Byte … H’10000 65536 Byte … H’20000 to H’3FFFF 131072 Byte

30.4.2.3 Interrupt status register (CXR21)

CXR21 indicates interrupt status.

Read restrictions:

When the PHY used does not have an INT signal, consider the read value of PHYI bit (bit3) as undefined.

Write restrictions:

Writing 0 is invalid. 1 or 0 cannot be written to bit 3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFRI	PHYI	LINKI	—	FCI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	X	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PFRI	0b	R/W	PAUSE Frame Retry Interrupt Interrupt source based on the PAUSE frame transmission retry count: Indicates that the PAUSE frame transmission is retried for the number of times specified using the CXR81 register. This bit can be cleared to 0 in the following way. (1) Read the CXR82 register to clear CXR82. (2) Write 1 to this bit.
3	PHYI	Xb	R/W	PHY Interrupt PHY chip interrupt source: Indicates status of 'PHY INT' of the E-MAC input/output signals. 1: Interrupt from PHY is asserted. 0: Interrupt from PHY is negated. Signal polarity of 'PHY INT' can be set using CXR2C. *Note*: When the PHY used does not have an INT signal, the read value of this bit should be undefined.
2	LINKI	0b	R/W	LINK Interrupt Link status change interrupt source: Indicates that on/off of bit 0 (LINK) of CXR2B has changed. Write 1 to clear this bit. *Note*: Before referring to this bit, determine whether to use PHY_LINK input signal or RGMII-Inband Status LINK bit as the link using SEL_LINK (bit 0) in the CXR31 register.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
0	FCI	0b	R/W	False Carrier Interrupt Invalid carrier detected interrupt source: Indicates that an invalid carrier was detected in the PHY chip. Write 1 to clear this bit to 0.

30.4.2.4 Enables/disables interrupts register (CXR22)

CXR22 enables/disables interrupts.

It is possible to enable/disable interrupts for each interrupt source indicated in CXR21 (bit unit).

Write restrictions:

If the PHY used does not have an INT signal, set PHYIM bit (bit 3) to 0. In the case of without using PHY_LINK signal and without using RGMII Inband Status function (case of setting ‘10’ or ‘11’ to SEL_LINK of CXR31), set PHYIM bit (bit 2) to 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFRIM	PHYIM	LINKIM	—	FCIM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	PFRIM	0b	R/W	PAUSE Frame Retry Interrupt Mask 1: Interrupt by PFRI bit in the CXR21 register is enabled. 0: Interrupt by PFRI bit in the CXR21 register is disabled.
3	PHYIM	0b	R/W	PHY Interrupt Mask 1: Interrupt by PHYI bit in the CXR21 register is enabled. 0: Interrupt by PHYI bit in the CXR21 register is disabled. *Note*: If the PHY used does not have an INT signal, set this bit to 0.
2	LINKIM	0b	R/W	LINK Interrupt Mask 1: Interrupt by LINKI bit in the CXR21 register is enabled. 0: Interrupt by LINKI bit in the CXR21 register is disabled.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	FCIM	0b	R/W	False Carrier Interrupt Mask 1: Interrupt by FCI bit in the CXR21 register is enabled. 0: Interrupt by FCI bit in the CXR21 register is disabled.

30.4.2.5 PHY register access register (CXR23)

There are two methods to access the PHY register: using the E-MAC-MDIOC block and using the CXR23 register.

Which access method to be used can be set using the MDIOMOD register.

For PHY register access using CXR23, see **Section 30.5.14.3(2), PHY Register Access Method (Using CXR23)** and IEEE Standard 802.3, 2000 Edition “22.2.4.5 Management frame structure.”

Write restrictions:

Writing bit 3 is invalid.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MDI	MDO	MM	MDC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	X	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	MDI	Xb	R/W	Management Data Input
2	MDO	0b	R/W	Management Data Output
1	MM	0b	R/W	Management Data Management mode The write value should be 1. The read value should be 0.
0	MDC	0b	R/W	Management Clock

30.4.2.6 Link status register (CXR2B)

CXR2B indicates link status.

Write restrictions:

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SOFT_LINK	SIGNAL_LINK	INBAND_LINK	LINK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	SOFT_LINK	0b	R	Software LINK Indicates the LINK ON/OFF status of software LINK setting (CXR2G) 1: Link ON (Link Up) 0: Link OFF (Link Down) This bit indicates the value of CXR2G.
2	SIGNAL_LINK	0b	R	LINK signal input Indicates the LINK ON/OFF status of the PHY_LINK input signal. 1: Link ON (Link Up) 0: Link OFF (Link Down) *Note*: Before referring to this bit, set the polarity of the PHY_LINK input signal using PHYLINKP (bit 1) in the CXR2C register.
1	INBAND_LINK	Xb	R	Inband Status LINK Indicates the link bit of RGMII Inband status. 1: Link ON (Link Up) 0: Link OFF (Link Down) *Note*: If the PHY used does not support Inband-status, the read value of this bit should be undefined.
0	LINK	0b	R	LINK status Indicates link status. 1: Link ON (Link Up) 0: Link OFF (Link Down) *Note*: Before referring to this bit, determine whether to use the PHY_LINK input signal, RGMII-Inband Status LINK bit or software LINK as the link using SEL_LINK (bit3, bit0) in the CXR31 register. In addition, to use the PHY_LINK input signal, set the polarity of PHY_LINK input using PHYLINKP (bit 1) in the CXR2C register.

30.4.2.7 Polarity set register (CXR2C)

CXR2C sets the polarity of ‘PHY_LINK’ and ‘PHY_INT’ signals.

Write restrictions:

Rewrite to this register with bits 2 and 3 in CXR22 being 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PHYLINKP	PHYIP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	PHYLINKP	0b	R/W	PHY_LINK Polarity Sets polarity of input signal ‘avb_miimg_link’. 1: High active (LINK ON [LINK UP] status with high) 0: Low active (LINK ON [LINK UP] status with low)
0	PHYIP	0b	R/W	PHY Interrupt Polarity Sets polarity of input signal ‘avb_miimg_int’. 1: High active (interrupt status with high) 0: Low active (interrupt status with low)

30.4.2.8 In-Band Status set register (CXR31)

CXR31 sets whether to apply In-Band Status function*[1] of RGMII-IF to the E-MAC operating mode.

It is possible to select the configured value by software for Link status.

Note 1. In-Band Status function (optional) is a notification function from PHY to E-MAC about Link Status (Up/Down), transmission speed (10 M/100 M/1 G) and Duplex (Half/Full) via RGMII-IF.

Write restrictions:

Rewrite to this register with transmission function disabled (CXR20 TPE = 0) and reception function also disabled (CXR20 RPE = 0). If the PHY used does not support In-band Status, set ‘H’0000000’ to this register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SEL_LINK[1]	SEL_SPEED	SEL_DUPLEX	SEL_LINK[0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3	SEL_LINK[1]	0b	R/W	Select LINK See bit0 of this register.
2	SEL_SPEED	0b	R/W	Select Speed*[1, *2] ‘1’: The information from In-band Status is used for setting the E-MAC operation speed. ‘0’: CXR2D register value is used for setting the E-MAC operation speed.
1	SEL_DUPLEX	0b	R/W	Select Duplex*[1, *2] ‘1’: The information from In-band Status is used for setting Full/Half Duplex. ‘0’: CXR20 DPM bit value is used for setting Full/Half Duplex.
0	SEL_LINK[0]	0b	R/W	Select LINK*[1, *2] SEL_LINK [1:0] ‘11’: The information from CXR20 TPE bit value is used for setting the LINK.*[3] The status of CXR20 TPE bit is set to CXR2B LINK bit. When this value is set, set ‘0’ (disable LINK interrupt) to CXR22 LINKIM bit. ‘10’: The information from CXR2B SOFTWR_LINK bit value is used for setting the LINK.*[3] The status of CXR2G SOFT_LINK bit is set to CXR2B LINK bit. When this value is set, set ‘0’ (disable LINK interrupt) to CXR22 LINKIM bit. ‘01’: The information from In-band Status is used for setting the LINK. ‘00’: The information from the PHY_LINK input signal is used for setting the LINK.

Note 1. If the PHY does not support In-band Status is used or If MII-IF is used, do not set value using In-band Status.

Note 2. If the device driver software is developed as common for MII-IF and RGMII-IF, not using this register is recommended.

Note 3. When Low Power Mode E-MAC is used, it needs to know the status of PHY as Link Up (Link ON). But, when it does not have the information to judge the status of LINK (because MII-IF is used and PHY_LINK signal is not used), set ‘10’ or ‘11’.
In the case of SEL_LINK [1:0] = ‘10’, CXR2B SOFT_LINK is used. So, software has to check the LINK status by register of

PHY and set that information to CXR2G.
In the case of SEL_LINK [1:0] = ‘11’, CXR20 TPE is used as LINK status.

30.4.2.9 In-Band Status indicate register (CXR32)

CXR32 indicates In-Band-Status of RGMII-IF.

In-Band Status function (option) is the function that the PHY notifies the E-MAC of link status (Up/Down), transfer rate (10 M/100 M/1 G) and duplex (full or half duplex), using the RGMII interface signal.

Write restrictions:

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	INBAND_SPEED		INBAND_DUPLEX	INBAND_LINK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	X	X	X	X
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 2	INBAND_SPEED	XXb	R	In-Band Status Speed*[1] '10': Indicates that the speed of RGMII In-Band status is 1 Gbps. '01': Indicates that the speed of RGMII In-Band status is 100 Mbps. '00': Indicates that the speed of RGMII In-Band status is 10 Mbps.
1	INBAND_DUPLEX	Xb	R	In-Band Status Duplex*[1] '1': Indicates that the Duplex bit of RGMII In-Band status is Full Duplex. '0': Indicates that the Duplex bit of RGMII In-Band status is Half Duplex.
0	INBAND_LINK	Xb	R	In-Band Status LINK*[1] '1': Indicates that LINK of RGMII In-Band status is LINK ON (LINK UP). '0': Indicates that LINK of RGMII In-Band status is LINK OFF (LINK DOWN).

Note 1. If the PHY used does not support Inband-status, the read value of those bits should be undefined.

30.4.2.10 Mode indicate register (CXR33)

CXR33 indicates the speed, duplex, and link operating mode of the E-MAC.

Write restrictions:

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SPEED[1:0]		DUPLEX	LINK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 2	SPEED	00b	R	Speed ‘10’: Transfer rate mode of E-MAC is 1 Gbps. ‘01’: Transfer rate mode of E-MAC is 100 Mbps. ‘00’: Transfer rate mode of E-MAC is 10 Mbps.
1	DUPLEX	0b	R	Duplex ‘1’: Duplex mode of E-MAC is Full Duplex. ‘0’: Duplex mode of E-MAC is Half Duplex.
0	LINK	0b	R	LINK ‘1’: LINK status is Link On (Link Up). ‘0’: LINK status is Link Off (Link Down).

30.4.2.11 PHY interface select register (CXR35)

CXR35 selects RGMII or MII interface.

Write restrictions:

(1) After release reset, make write-access to this register before making write-access to other registers (except MDIOMOD). Even if not need to change the value of this register, make write-access to this register at least one time. Because RGMII/MII MODE is recognized by accessing this register.

(2) If need to re-write the value of this register after making write-access to other register, reset to E-MAC is needed.

(3) If need to re-write the value of this register without making write-access to other register, wait the quadruple time which is set at HALFCYC_CLKSW.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HALFCYC_CLKSW[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SEL_MODPOL	—	—	—	SEL_MODIN	—	—	—	—	—	—	SEL_XMII[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	HALFCYC_CLKSW	H'FFFF	R/W	A clock stop time at the clock switching The wait time (over 20 us) is needed when PHY interface is selected at initializing Gigabit Ethernet Interface. Set the number of clk_chi (APB clock) cycle to be half of wait time (over 10 us). H'0000: 1 cycle of clk_chi H'0001: 2 cycle of clk_chi H'0002: 3 cycle of clk_chi … H'03E8: 1000 cycle of clk_chi (If clk_chi is 100 MHz (10 ns), half of wait time will be 10 us) H'07D0: 2000 cycle of clk_chi (If clk_chi is 200 MHz (5 ns), half of wait time will be 10 us) … H'FFFF: 65536 cycle of clk_chi (default value)
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	SEL_MODPOL	0b	R/W	polar selection of RGMII/MII Mode input signal '0': (default value) E-MAC recognizes RGMII-IF is used. '1': E-MAC recognizes MII-IF is used.
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	SEL_MODIN	0b	R/W	method for selection of RGMII/MII '0': SEL_XMII field of this register is used for RGMII/MII selection (default value) '1': Reserved
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1 to 0	SEL_XMII	00b	R/W	Selection of PHY interface to be used ‘00’: RGMII interface is used (default value) ‘01’: Reserved ‘10’: MII interface is used ‘11’: Reserved *Note*: The case of SEL_MODIN (bit8) = ‘0’, this field is used to select PHY interface. The case of SEL_MODIN (bit8) = ‘1’, this field is not used and Mode signals which is provided from out of E-MAC is used to select PHY interface.

Note: This register needs to write access at least one time.

[A] The case which CXR35 SEL_XMII is used for the selection of RGMII/MII in APB Clock 100 MHz.

(1) To use RGMII interface, Set ‘H’03E8 0000’ to this register.

(2) To use MII interface, Set ‘H’03E8 0002’ to this register.

30.4.2.12 PHY interface indicate register (CXR36)

CXR36 indicates the status of RGMII/MII Interface.

Write restrictions:

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STS_XMII[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1 to 0	STS_XMII	00b	R	Status of selected PHY interface ‘00’: RGMII interface ‘01’: Reserved ‘10’: MII interface ‘11’: Reserved

30.4.2.13 PAUSE frame register 1 (CXR71)

CXR71 sets the timer value for Auto PAUSE frame.

Write restrictions:

Set a value other than all 0 when transmission system flow control is enabled. (CXR20_bit16 = 1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APFTP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	APFTP	H’0000	R/W	Auto Pause Frame Time Parameter The value set in this register is used as the Timer value for PAUSE frame during Auto PAUSE Frame transmission. *Note*: Set the value of this register to other than 0000 when transmission system flow control is enabled (CXR20_bit 16 = 1). <set value> <bit time> H’0000 — H’0001 512 × 1 bit time H’0002 512 × 2 bit time H’0003 512 × 3 bit time … H’FFFF 512 × 65535 bit time <transfer rate> <1 bit time> 1000 Mbps 1 ns 100 Mbps 10 ns 10 Mbps 100 ns

30.4.2.14 PAUSE frame register 2 (CXR72)

CXR72 starts the transmission of Manual PAUSE frames and sets the timer value for Manual PAUSE frame.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPFTP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	MPFTP	H’0000	R/W	Manual Pause Frame Time Parameter When a value is written to this register, PAUSE frame transmission is started using the value as the timer value. <set value> <bit time> H’0000 — H’0001 512 × 1 bit time H’0002 512 × 2 bit time H’0003 512 × 3 bit time … H’FFFF 512 × 65535 bit time <transfer rate> <1 bit time> 1000 Mbps 1 ns 100 Mbps 10 ns 10 Mbps 100 ns

30.4.2.15 PAUSE frame register 3 (CXR8A)

CXR8A is a transmit counter for PAUSE frame.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFTXC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFTXC	H'0000	R/W	Pause Frame TX Counter Indicates the number of times of transmission of Auto PAUSE frames and Manual PAUSE frames in total. This register is cleared to 0 when read.

30.4.2.16 PAUSE frame register 4 (CXR80)

CXR80 is a receive counter for PAUSE frame.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRXC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFRXC	H'0000	R/W	Pause Frame RX Counter Indicates the number of PAUSE frames received when receive system flow control is enabled (CXR20_bit17 = 1). This register is cleared to 0 when read.

30.4.2.17 PAUSE frame register 5 (CXR81)

CXR81 sets the upper limit of the number of autonomous transmissions of PAUSE frame.

Write restrictions:

Rewriting this register is prohibited when 1 is set to CXR20_bit 5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRTULMT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFRTULMT	H’0000	R/W	Pause Frame Retry Upper LiMiT Sets the number of times of Auto PAUSE frame transmission (upper limit) from the time of asserting “Reception FIFO Critical Level Notification” from DMAC assertion to negation. <set value> <Number of transmissions> H’0000 No limit H’0001 1 time H’0002 2 times H’0003 3 times … H’FFFF 65535 times

30.4.2.18 PAUSE frame register 6 (CXR82)

CXR82 counter indicates the number of Auto PAUSE frame transmissions from “Reception FIFO Critical Level Notification” from DMAC assertion to negation.

Write restrictions:

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRTC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PFRTC	H’FFFF	R	Pause Frame Retry Counter Indicates the number of times of Auto PAUSE frame transmission from “Reception FIFO Critical Level Notification” from DMAC assertion to negation. This register is cleared to 0 when read and at the start of “Reception FIFO Critical Level Notification” from DMAC assertion.

30.4.2.19 E-MAC operating mode register 2 (CXR2D)

CXR2D sets the operating mode of the E-MAC.

Rewriting this register bits is prohibited with transmission function enabled (CXR20<TPE> = 1) and reception function enabled (CXR20<RPE> = 1).

Write restrictions:

Rewriting this register is prohibited when 1 is set to CXR20_bit 5 or 6.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SPEED[1:0]		—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31 to 6</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.</td></tr>
<tr><td>5 to 4</td><td>SPEED</td><td>00b</td><td>R/W</td><td>Transmit SPEED
Set transfer rate.
<table>
<tr><th>SPEED[1]
bit5</th><th>SPEED[0]
bit4</th><th>Transmit Speed</th></tr>
<tr><td>0</td><td>0</td><td>10 Mbps</td></tr>
<tr><td>0</td><td>1</td><td>100 Mbps</td></tr>
<tr><td>1</td><td>0</td><td>1000 Mbps</td></tr>
<tr><td>1</td><td>1</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>3 to 0</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.</td></tr>
</table>

30.4.2.20 Software LINK status register (CXR2G)

CXR2G indicates the status of Software LINK.

Write restrictions:

> Before setting ‘1’ to this register, set SEL_LINK of CXR31 whether using PHY_LINK input signal, using LINK bit of RGMII-Inband Status or using LINK setting by software.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SOFT_LINK	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	SOFT_LINK	0b	R/W	Software LINK This bit is used when the LINK status is set by software. ‘1’: Link ON (Link Up) ‘0’: Link OFF (Link Down) *Note*: Before setting ‘1’ to this bit, set SEL_LINK of CXR31 whether using PHY_LINK input signal, using LINK bit of RGMII-Inband Status or using LINK setting by software. Additionally, if PHY_LINK input signal is used, set the polar of PHY_LINK input signal by PHYLINKP of CXR2C.
2 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

30.4.2.21 Mac Address register 1 (CXR24)

CXR24 set the universal MAC address (upper 32 bits).

Write restrictions:

Rewriting this register is prohibited when 1 is set to CXR20_bit 5 or 6.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UMADR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UMADR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	UMADR	H'0000_0000	R/W	Upper MAC Address Upper-32bit MAC address

30.4.2.22 Mac Address register 2 (CXR25)

CXR25 set the universal MAC address (lower 16 bits).

Write restrictions:

Rewriting this register is prohibited when 1 is set to CXR20_bit 5 or 6.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LMADR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	LMADR	H'0000	R/W	Lower MAC Address Lower-16bit MAC address

30.4.2.23 TINT1 counter register (CXR40)

CXR40 indicates the number of occurrences of TINT1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TINT1_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TINT1_CNT	H’0000	R/W	TINT1 Counter Indicates the number of occurrences of TINT1. TINT1: Frame transmission timeout When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.24 TINT2 counter register (CXR41)

CXR41 indicates the number of occurrences of TINT2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TINT2_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TINT2_CNT	H’0000	R/W	TINT2 Counter Indicates the number of occurrences of TINT2. TINT2: Delayed collision detection When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.25 TINT3 counter register (CXR42)

CXR42 indicates the number of occurrences of TINT3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TINT3_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TINT3_CNT	H’0000	R/W	TINT3 Counter Indicates the number of occurrences of TINT3. TINT3: Transmission is terminated in error with TINT3 generated under the following conditions: Delay collision occurs at the end of transmit frame, the receive frame at the time of the collision continues until the start of next frame transmission, but does not continue until the end of the transmission. However, these conditions do not occur in normal operation.

30.4.2.26 RINT1 counter register (CXR50)

CXR50 indicates the number of occurrences of RINT1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT1_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT1_CNT	H’0000	R/W	RINT1 Counter Indicates the number of occurrences of RINT1. RINT1: Receive frame CRC error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.27 RINT2 counter register (CXR51)

CXR51 indicates the number of occurrences of RINT2.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT2_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT2_CNT	H’0000	R/W	RINT2 Counter Indicates the number of occurrences of RINT2. RINT2: Frame reception error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.28 RINT3 counter register (CXR52)

CXR52 indicates the number of occurrences of RINT3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT3_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT3_CNT	H’0000	R/W	RINT3 Counter Indicates the number of occurrences of RINT3. RINT3: Erroneous frame length (less than 64 bytes frame was received.) When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.29 RINT4 counter register (CXR53)

CXR53 indicates the number of occurrences of RINT4.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT4_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT4_CNT	H’0000	R/W	RINT4 Counter Indicates the number of occurrences of RINT4. RINT4: Erroneous frame length {A frame longer than the maximum frame length was received. Maximum frame length (including CRC) can be specified using the LEN_LMT field in the CXR2A register.} When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.30 RINT5 counter register (CXR54)

CXR54 indicates the number of occurrences of RINT5.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT5_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT5_CNT	H’0000	R/W	RINT5 Counter Indicates the number of occurrences of RINT5. RINT5: Fractional bit error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data)

30.4.2.31 RINT6 counter register (CXR55)

CXR55 indicates the number of occurrences of RINT6.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT6_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT6_CNT	H’0000	R/W	RINT6 Counter Indicates the number of occurrences of RINT6. RINT6: Carrier extension lost When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.32 RINT7 counter register (CXR56)

CXR56 indicates the number of occurrences of RINT7.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT7_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT7_CNT	H’0000	R/W	RINT7 Counter Indicates the number of occurrences of RINT7. RINT7: Carrier extension error When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.33 RINT8 counter register (CXR57)

CXR57 indicates the number of occurrences of RINT8.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RINT8_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RINT8_CNT	H’0000	R/W	RINT8 Counter Indicates the number of occurrences of RINT8. RINT8: Multicast frame received When CXR20_bit 26 = 1, this register is cleared to 0 when read. When CXR20_bit 26 = 0, this register is cleared to 0 when written to. (Cleared to 0 when written to, irrespective of the write data.)

30.4.2.34 MDIO status register (MDIOSTS)

MDIOSTS indicates the status of PHY register access (MDIO access).

Write restrictions:

Write disabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BSY
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	BSY	0b	R	PHY register access Busy flag 0: Access enabled 1: Busy Do not write to MDIOCMD, MDIOADR, MDIODAT or MDIOMOD registers when this bit is1.

30.4.2.35 MDIO command register (MDIOCMD)

MDIOCMD specifies the commands for PHY register access (MDIO access).

Write restrictions:

Do not write to this register when the BSY bit in the MDIOSTS register is 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	OP	00b	R/W	Operation Code for PHY register access 00: Setting prohibited 01: WRITE 10: READ 11: Setting prohibited

30.4.2.36 MDIO address register (MDIOADR)

MDIOADR specifies PHY address and PHY register address.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	A[9:5]					A[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9 to 5	A[9:5]	00000b	R/W	PHY address 5 bits
4 to 0	A[4:0]	00000b	R/W	PHY register address 5 bits

30.4.2.37 MDIO data register (MDIODAT)

MDIODAT specifies PHY write data and indicates read data.

Write restrictions:

Do not write to this register when the BSY bit in the MDIOSTS register is 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	D[15:0]	H’0000	R/W	PHY register data When writing: Write data When reading: Read data

30.4.2.38 MDIO mode register (MDIOMOD)

MDIOMOD sets the operating mode of the MDIO controller.

Write restrictions:

Writing to this bit is prohibited during PHY register access.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SEL_MDIO_N	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CYC_OCLK[7:0]							
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	SEL_MDIO_N	1b	R/W	Selection for PHY register access (MDIO access selection) 0: Uses MDIO controller for PHY register access. 1: Uses CXR23 register for PHY register access. (Initial value)
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	CYC_OCLK	H’FF	R/W	MDC (MDIO control clock) half cycle (number of cycles for system clock) H’00: 7 cycles H’01: 7 cycles ... H’06: 7 cycles H’07: 8 cycles H’08: 9 cycles H’09: 10 cycles ... H’FE: 255 cycles H’FF: 256 cycles (Initial value) MDC half cycle should be set to be 200 ns or more. (Calculation example) 27 cycles at a 133-MHz Clock (1 cyc=7.5 ns), 200 ns or more -> Setting value is H’1A (26d) 7.5 × 27 = 202.5 ≥ 200 (Setting example) //66.66 MHz (15 ns): Setting value ≥ H’0D (13d) (14 cyc × 15 ns = 210 ns) //133.3 MHz (7.5 ns): Setting value ≥ H’1A (26d) (27 cyc × 7.5 ns=202.5 ns) //266.6 MHz (3.75 ns): Setting value ≥ H’35 (53d) (54 cyc × 3.75 ns=202.5 ns)

30.4.2.39 Low Power Mode register 1 (LPTXMOD1)

LPTXMOD1 sets the operating mode of the Low Power Transmission.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’ and CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	GCYC_TCSTOP[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MCYC_TCSTOP[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	GCYC_TCSTOP	H’000	R/W	Number of minimum clock cycles until stopping RGMII TXC (1 Gbps) 0: 10 cycles (Initial value) 1 to 8: (Setting prohibited) 9: 9 cycles 10: 10 cycles 11: 11 cycles (skip the rest)
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	MCYC_TCSTOP	H’000	R/W	Number of minimum clock cycles until stopping RGMII TXC (100 Mbps) 0: 10 cycles (Initial value) 1 to 8: (Setting prohibited) 9: 9 cycles 10: 10 cycles 11: 11 cycles (skip the rest)

30.4.2.40 Low Power Mode register 2 (LPTXMOD2)

LPTXMOD2 sets the operating mode of the Low Power Transmission.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’ and CXR20_bit5 = ‘0’. When set ‘1’ to this register, check the configuration of Clock stoppable of PHY is ‘1’.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	MAC_TLPI_TYPE	—	—	—	STP_TXC_LPI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	MAC_TLPI_TYPE	0b	R/W	Transmission LPI Operation Type of MAC ‘1’: MAC TX LPI Slave Mode E-MAC does not transmit LPI code during LPI period. But using LPI Wakeup parameters (LPTXGTH3/LPTXMTH3) to make the delay time of transmitting such as for PAUSE Frame. When making PHY control the operation of LPI (Low Power Idle), set ‘1’ to this bit. ‘0’: MAC TX LPI Master Mode E-MAC controls the operation of LPI. (Initial value)
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STP_TXC_LPI	0b	R/W	Setting of stopping LPI TXC (this bit is used only RGMII Mode) ‘1’: The clock for RGMII transmitting (fet_miitx_gtx_clk) is stopped during output of LPI code. ‘0’: The clock for RGMII transmitting (fet_miitx_gtx_clk) is not stopped during output of LPI code. (Initial value) Set ‘1’ to this bit when TX Clock stop capable (Note1) of PHY is ‘1’ *Note 1.* See the Bit 6 Clock Stop Capable (MAC may stop clock during LPI) of PCS Status register in PHY MMD extended register (MMD address 3, REG address 1) *Note 2.* In the case of CCC.ERCS = 1 (The “clk_miitx_gtx_refclk” stop function is enable), even if this bit was set to ‘0’, the clock for RGMII transmitting (fet_miitx_gtx_clk) is stopped during the stopping period of “clk_miitx_gtx_refclk”. Because “fet_miitx_gtx_clk” is generated from “clk_miitx_gtx_refclk”.

30.4.2.41 RGMII Low Power parameter register 1 (LPTXGTH1)

LPTXGTH1 sets the change timing to Low Power Idle Transmission Mode at 1 Gbps.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_GSTH1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_GSTH1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GSTH1	H’0000_0000	R/W	Threshold value of LPI code transmission standby status (1 Gbps) (Unit is 32 ns) 0: (Initial value) When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. 1 to 7: Setting prohibited ... 32: When MAC does not receive transmission request during 1024 ns. MAC starts transmitting LPI code. ... 31250: When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF.

30.4.2.42 RGMII Low Power parameter register 2 (LPTXGTH2)

LPTXGTH2 sets the change timing to Low Power Idle Transmission Mode after enabled function of transmitting LPI code at 1Gbps.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_GSTH2[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_GSTH2[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GSTH2	H’0000_0000	R/W	Threshold value of starting LPI code transmission standby status (1 Gbps) (Unit is 32 ns) 0: (Initial value) MAC will be ready for transmitting LPI code after 1.01 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. 1 to 7: Setting prohibited ... 32: MAC will be ready for transmitting LPI code after 1024 ns later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 31250000: MAC will be ready for transmitting LPI code after 1 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 31562500: MAC will be ready for transmitting LPI code after 1.01 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 100000000: MAC will be ready for transmitting LPI code after 3.2 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value In the standard, the behavior of PHY is undefined if PHY receives LPI code within 1 s after Link Up. Therefore, the recommended value was determined as 1.01 s. (1 s plus 1% margin) The following is reason why “LPM bit of CXR20 is set to ‘1’” was include to the condition. Internal Clock (32 ns:1 Gbps, 320 ns:100 Mbps) is used to measure the time after Link Up. But Internal Clock will change during transmission rate is un-determined. So, it is difficult to measure the time after Link Up exactly. Therefore, “LPM bit of CXR20 is set to ‘1’” was include to the condition as determined point of transmission rate of Low Power Mode.

30.4.2.43 RGMII Low Power parameter register 3 (LPTXGTH3)

LPTXGTH3 sets the operation return from Low Power Idle Transmission Mode at 1 Gbps.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value). Do not set 262144 (H’00040000) or more to this register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_GRTH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_GRTH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GRTH	H'0000_0000	R/W	Transmission Low Power resume time (1 Gbps) (Unit 32 ns) 0: (Initial value) After received transmit request, MAC transmits Normal Idle code during 17.184 us. And then MAC returns to Normal Mode from Low Power Mode. 1 to 7: Setting prohibited ... 32: After received transmit request, MAC transmits Normal Idle code during 1024 ns And then MAC returns to Normal Mode from Low Power Mode. ... 516: After received transmit request, MAC transmits Normal Idle code during 16.512 us. And then MAC returns to Normal Mode from Low Power Mode. 521: After received transmit request, MAC transmits Normal Idle code during 16.672 us. And then MAC returns to Normal Mode from Low Power Mode. 532: After received transmit request, MAC transmits Normal Idle code during 17.024 us. And then MAC returns to Normal Mode from Low Power Mode. 537: After received transmit request, MAC transmits Normal Idle code during 17.184 us. And then MAC returns to Normal Mode from Low Power Mode. ... 1000: After received transmit request, MAC transmits Normal Idle code during 17.184 us. And then MAC returns to Normal Mode from Low Power Mode. ... 262144 or more: Reserved (Setting prohibited) (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value The basic value of this field is LPI WakeUp Time (Tw_sys_tx) with 1% margin. In the case of 1000BASE-T, Tw_sys_tx is 16.5 us. But some of PHY needs 1 us for Wakeup Time. Therefore, recommended value was determined as 17.184 us. The setting value of this field will be the overhead time return from Low Power Mode. Do not set the value will be under the 16.5 us (Minimum Wakeup Time which is defined by standard) to this field. In addition, if PHY has the LPI transmission control function, PHY also make overhead time same as Wake-Up Time.

30.4.2.44 RGMII Low Power parameter register 4 (LPTXGTH4)

LPTXGTH4 sets the minimum period for Low Power Idle Transmission Mode at 1 Gbps.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_GCTH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_GCTH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_GCTH	H’0000_0000	R/W	Minimum Transmission Low Power time (1 Gbps) (Unit is 32 ns) 0: (Initial value) After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 1024 ns. 1 to 7: Setting prohibited 8: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 256 ns. ... 32: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 1024 ns. ... 6469: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 207.008 us (Sleep Time + α). ... 10000: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 320 us. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value Recommended value of this field is 1 us (1024 ns) for 1000BASE-T. This value based on followings: [1] This value should be sufficiency longer than 72 ns [2] This value should be sufficiency shorter than Wake Up Time (16.5 us). And selected the value which is close to geometric mean of [1] and [2].

30.4.2.45 MII Low Power parameter register 1 (LPTXMTH1)

LPTXMTH1 sets the change timing to Low Power Idle Transmission Mode at 100 Mbps.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_MSTH1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_MSTH1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MSTH1	H’0000_0000	R/W	Threshold value of LPI code transmission standby status (100 Mbps) (Unit is 320 ns) 0: (Initial value) When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. 1 to 7: Setting prohibited 8: When MAC does not receive transmission request during 10240 ns, MAC starts transmitting LPI code. ... 3125: When MAC does not receive transmission request during 1 ms, MAC starts transmitting LPI code. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF.

30.4.2.46 MII Low Power parameter register 2 (LPTXMTH2)

LPTXMTH2 sets the change timing to Low Power Idle Transmission Mode after enabled function of transmitting LPI code at 100 Mbps.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_MSTH2[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_MSTH2[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MSTH2	H’0000_0000	R/W	Threshold value of starting LPI code transmission standby status (100 Mbps) (Unit is 320 ns) 0: (Initial value) MAC will be ready for transmitting LPI code after 1.01 s later from Link Up or LPM bit of CXR20 is set to ‘1’. 1 to 7: Setting prohibited ... 32: MAC will be ready for transmitting LPI code after 10240 ns later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 3125000: MAC will be ready for transmitting LPI code after 1.0 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 3156250: MAC will be ready for transmitting LPI code after 1.01 s from LinkUp/LPM bit of CXR20 is set to ‘1’. ... 10000000: MAC will be ready for transmitting LPI code after 3.2 s later from LinkUp/LPM bit of CXR20 is set to ‘1’. ... (skip the rest) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value In the standard, the behavior of PHY is undefined if PHY receives LPI code within 1 s after Link Up. Therefore, the recommended value was determined as 1.01 s. (1 s plus 1% margin) The following is reason why “LPM bit of CXR20 is set to ‘1’” was include to the condition. Internal Clock (32 ns:1 Gbps, 320 ns:100 Mbps) is used to measure the time after Link Up. But Internal Clock will change during transmission rate is un-determined. So, it is difficult to measure the time after Link Up exactly. Therefore, “LPM bit of CXR20 is set to ‘1’” was include to the condition as determined point of transmission rate of Low Power Mode.

30.4.2.47 MII Low Power parameter register 3 (LPTXMTH3)

LPTXMTH3 sets the operation return from Low Power Idle Transmission Mode at 100 Mbps.

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value). Do not set 262144 (H’00040000) or more to this register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_MRTH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_MRTH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MRTH	H’0000_0000	R/W	Transmission Low Power resume time (100 Mbps) (Unit 320 ns) 0: (Initial value) After received transmit request, MAC transmits Normal Idle code during 30.40 us. And then MAC returns to Normal Mode from Low Power Mode. 1 to 7: Setting prohibited ... 32: After received transmit request, MAC transmits Normal Idle code during 10240 ns And then MAC returns to Normal Mode from Low Power Mode. ... 94: After received transmit request, MAC transmits Normal Idle code during 30.08 us And then MAC returns to Normal Mode from Low Power Mode. 95: After received transmit request, MAC transmits Normal Idle code during 30.40 us And then MAC returns to Normal Mode from Low Power Mode. ... 262144 or more: Reserved (Setting prohibited) *Note*: This value is used by internal logic and does not mean indicating accurate time on PHY IF. Recommended setting value The basic value of this field is LPI WakeUp Time (Tw_sys_tx) with 1% margin. In the case of 100BASE-TX, Tw_sys_tx is 30 us. Therefore, recommended value was determined as 30.4 us. The setting value of this field will be the overhead time return from Low Power Mode. Do not set the value under the 30 us (Minimum Wakeup Time which is defined by standard) to this field.

30.4.2.48 MII Low Power parameter register 4 (LPTXMTH4)

LPTXMTH4 sets the minimum period for Low Power Idle Transmission Mode at 100 Mbps

Write restrictions:

> Write to this register when the CXR20_bit6 = ‘0’, CXR20_bit5 = ‘0’. Recommended value is H’00000000 (Initial value).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPTX_MCTH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPTX_MCTH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LPTX_MCTH	H’0000_0000	R/W	Minimum Transmission Low Power time (100 Mbps) (Unit is 320 ns) 0: (Initial value) After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 3200 ns. 1 to 7: Setting prohibited 8: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 2560 ns. ... 10: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 3200 ns. ... 844: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 270.08 us (Sleep Time+α). ... 1000: After started to output Transmission LPI code, MAC will continue to output Transmission LPI code at least 320 us. ... (skip the rest) *Note*: This value is used by internal logic and doesn’t mean indicating accurate time on PHY IF. Recommended setting value Recommended value of this field is 3.2 us (3200 ns) for 100BASE-TX. This value based on followings: [1] his value should be sufficiency longer than 360 ns [2] This value should be sufficiency shorter than Wake Up Time (30 us). And selected the value which is close to geometric mean of [1] and [2].

30.4.3 TOE Registers

30.4.3.1 Checksum operating mode register (CSR0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FIFOCAP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RBP	TBP	—	—	RPE	TPE	—	—	—	CCM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17 to 16	FIFOCAP	00b	R/W	FIFO CAPACITY TOE FIFO Capacity 00b: 2 Kbyte (512 Word × 32bit) 01b: 4 Kbyte (1024 Word × 32bit) 10b: 8 Kbyte (2048 Word × 32bit) 11b: 16 Kbyte (4096 Word × 32bit) Write to this field when TPE is ‘0’.
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	RBP	0b	R/W	Receive Bypass Enable Bypass Enable of TOE Rx function. 0b: Rx Function is not bypassed. 1b: Rx Function is bypassed. (for debug) *Note*: This bit is used only for debug. Do not set 1b during normal operation. When this bit is set to “1", all Rx Functions of TOE are bypassed. Therefore, TOE sends Rx frame from E-MAC to DMAC even if RPE is ‘0’. Write to this bit when RPE is ‘0’. When this bit is set to “0", set E-MAC as CRC Pass Through Mode (CXR20 RCPT = “1”). When this bit is set to “1”, set E-MAC as Non-CRC Pass Through Mode (CXR20 RCPT = “0”).
8	TBP	0b	R/W	Transmit Bypass Enable Bypass Enable of TOE Tx Function 0b: Tx Function is not bypassed. 1b: Tx Function is bypassed (for debug) *Note*: This bit is used only for debug. Do not set 1b during normal operation. When this bit is set to “1", all Tx Functions of TOE are bypassed. Therefore, TOE sends Tx frame from DMAC to E-MAC even if TPE is ‘0’. Since Tx Frame does not go through TOE FIFO, there is no latency. Write to this bit when TPE is ‘0’.
7 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
5	RPE	0b	R/W	Receive Port Enable Enable of Frame Output Function to DMAC. 0b: Output Function is disabled. 1b: Output Function is enabled. When value is changed from ‘1’ to ‘0’ during TOE is outputting any frames, the value of this bit will be ‘0’ after finished outputting. Therefore, ‘1’ is read as read value during TOE is outputting frames.
4	TPE	0b	R/W	Transmit Port Enable Enable of Frame Output Function to E-MAC. 0b: Output Function is disable. 1b: Output Function is enabled. Procedure of setting ‘0’ to this bit. (1) Set ‘0’ to TPE. (2) Continue to read TPE until it will be read as ‘0’. (TPE will be ‘0’ when TOE FIFO will be empty) (3) If read value is ‘0’, procedure of setting ‘0’ is finished.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CCM	0b	R/W	Counter Clear Mode This bit designates the clear method of Interrupt Status Counters (CSR20, CSR30, CSR31, CSR32) 1b: When each Counter Registers are read, they are cleared to ‘0’. 0b: When each Counter Registers are written any value, they are cleared to ‘0’. Write to this bit when TPE is ‘0’ and RPE is ‘0’.

30.4.3.2 Tx Frame Checksum Enable register (CSR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	TDHD	TAHD	TROUT	THOP	—	TICMP6	TUDP6	TTCP6	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TICMP4	TUDP4	TTCP4	—	—	—	TIP4
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	TDHD	0b	R/W	Transmit Destination Option Header Enable When the frame from DMAC has a Destination Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are '0', the configuration of this bit is ignored. Write to this bit when TPE is '0'.
26	TAHD	0b	R/W	Transmit Authentication Header Enable When the frame from DMAC has an AH header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are '0', the configuration of this bit is ignored. Write to this bit when TPE is '0'.
25	TROUT	0b	R/W	Transmit Routing Header Enable When the frame from DMAC has a Routing header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are '0', the configuration of this bit is ignored. Write to this bit when TPE is '0'.
24	THOP	0b	R/W	Transmit Hop-by-Hop Option Header Enable When the frame from DMAC has a Hop-by-Hop Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are '0', the configuration of this bit is ignored. Write to this bit when TPE is '0'.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	TICMP6	0b	R/W	Transmit ICMP Checksum in IPv6 Enable Enable of IPv6 ICMP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable When this bit is '1', the value of bits [27:24] are valid. Write to this bit when TPE is '0'.

Bit	Bit Name	Initial Value	R/W	Description
21	TUDP6	0b	R/W	Transmit UDP Checksum in IPv6 Enable Enable of IPv6 UDP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when TPE is ‘0’.
20	TTCP6	0b	R/W	Transmit TCP Checksum in IPv6 Enable Enable of IPv6 TCP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when TPE is ‘0’.
19 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	TICMP4	0b	R/W	Transmit ICMP Checksum in IPv4 Enable Enable of IPv4 ICMP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.
5	TUDP4	0b	R/W	Transmit UDP Checksum in IPv4 Enable Enable of IPv4 UDP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.
4	TTCP4	0b	R/W	Transmit TCP Checksum in IPv4 Enable Enable of IPv4 TCP Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	TIP4	0b	R/W	Transmit IPv4 Checksum Enable Enable of IPv4 Header Checksum Calculation in the frame from DMAC. 0b: Disable 1b: Enable Write to this bit when TPE is ‘0’.

30.4.3.3 Rx Frame Checksum Enable register (CSR2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	RDHD	RAHD	RROUT	RHOP	—	RICMP6	RUDP6	RTCP6	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RICMP4	RUDP4	RTCP4	—	—	—	RIP4
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	RDHD	0b	R/W	Receive Destination Option Header Enable When the frame from E-MAC has a Destination Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
26	RAHD	0b	R/W	Receive Authentication Header Enable When the frame from E-MAC has an AH header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
25	RROUT	0b	R/W	Receive Routing Header Enable When the frame from E-MAC has a Routing header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
24	RHOP	0b	R/W	Receive Hop-by-Hop Option Header Enable When the frame from E-MAC has a Hop-by-Hop Option header, this bit configures whether TOE ascribes this frame to Supported Frame or not. 0b: Ascribes to Unsupported Frame 1b: Ascribes to Supported Frame If bits 20, 21 and 22 of this register are ‘0’, the configuration of this bit is ignored. Write to this bit when RPE is ‘0’.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	RICMP6	0b	R/W	Receive ICMP Checksum in IPv6 Enable Enable of IPv6 ICMP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when RPE is ‘0’.

Bit	Bit Name	Initial Value	R/W	Description
21	RUDP6	0b	R/W	Receive UDP Checksum in IPv6 Enable Enable of IPv6 UDP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when RPE is ‘0’.
20	RTCP6	0b	R/W	Receive TCP Checksum in IPv6 Enable Enable of IPv6 TCP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable When this bit is ‘1’, the value of bits [27:24] are valid. Write to this bit when RPE is ‘0’.
19 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	RICMP4	0b	R/W	Receive ICMP Checksum in IPv4 Enable Enable of IPv4 ICMP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is ‘0’.
5	RUDP4	0b	R/W	Receive UDP Checksum in IPv4 Enable Enable of IPv4 UDP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is ‘0’.
4	RTCP4	0b	R/W	Receive TCP Checksum in IPv4 Enable Enable of IPv4 TCP Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is ‘0’. These bits are read as 0. The write value should be always 0.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RIP4	0b	R/W	Receive IPv4 Checksum Enable Enable of IPv4 Header Checksum Calculation in the frame from E-MAC. 0b: Disable 1b: Enable Write to this bit when RPE is ‘0’.

30.4.3.4 Tx Extended Header Number register (CSR3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TNUEXHED[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	TNUEXHED	H’8	R/W	Transmit Support Number of Extension Header Transmit Support Number of Extension Header in the frame from DMAC H’0: 16 H’1: 1 … H’E: 14 H’F: 15 If the number of Extension Header in the IPv6 frame is over the configuration of this field, TOE ascribes this frame to unsupported. Write to this field when TPE is ‘0’.

30.4.3.5 Rx Extended Header Number register (CSR4)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RNUEXHED[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	RNUEXHED	H'8	R/W	Receive Support Number of Extension Header Receive Support Number of Extension Header in the frame from E-MAC. H'0: 16 H'1: 1 … H'E: 14 H'F: 15 If the number of Extension Header in the IPv6 frame is over the configuration of this field, TOE ascribes this frame to unsupported. Write to this field when RPE is '0'.

30.4.3.6 Unsupported Tx Frame Counter register (CSR20)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TNSCNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	TNSCNT	H’0000	R/W	Transmit Not Support Frame Counter Counter for Transmitted Unsupported Frames When CSR0_bit0 is ‘1’, this register is cleared to ‘0’ by read access. When CSR0_bit0 is ‘0’, this register is cleared to ‘0’ by write access of any data. The condition of counting up is OR of followings. ● TYPE/TPID is Unsupported ● Not (“MF = 0” and “Fragment Offset = H’0000”) ● Protocol is Unsupported ● Next Header of IPv6 frame is Unsupported ● The number of Next Header in IPv6 frame is 9 or more ● Checksum calculation for a particular frame is disabled by CSR1 ● The Frame Length is shorter than the result of Frame Analysis by TOE.

30.4.3.7 Unsupported Rx Frame Counter register (CSR30)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RNSCNT[15:8]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RNSCNT	H’0000	R/W	Receive Not Support Frame Counter Counter for Received Unsupported Frames When CSR0_bit0 is ‘1’, this register is cleared to ‘0’ by read access. When CSR0_bit0 is ‘0’, this register is cleared to ‘0’ by write access of any data. The condition of counting up is OR of followings. ● TYPE/TPID is Unsupported ● Not (“MF = 0” and “Fragment Offset = H’0000”) ● Protocol is Unsupported ● Next Header of IPv6 frame is Unsupported ● The number of Next Header in IPv6 frame is 9 or more ● Checksum calculation for a particular frame is disabled by CSR2 ● The Frame Length is shorter than the result of Frame Analysis by TOE.

30.4.3.8 Rx IPv4 Header checksum Error Counter register (CSR31)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R4ECNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	R4ECNT	H’0000	R/W	Receive IPv4 Header Checksum Error Counter Counter for the number of IPv4 Header Checksum Error in the frames from E-MAC. When CSR0_bit0 is ‘1’, this register is cleared to ‘0’ by read access. When CSR0_bit0 is ‘0’, this register is cleared to ‘0’ by write access of any data. The condition of counting up is following. When the frame which has IPv4 Checksum Error comes, TOE counts up this register.

30.4.3.9 Rx TCP/UDP/ICMP checksum Error Counter register (CSR32)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTUECNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	RTUECNT	H’0000	R/W	Receive TCP/UDP/ICMP Checksum Error Counter Counter for the number of TCP/UDP/ICMP Checksum Error in the frames from E-MAC. When CSR0_bit0 is ‘1’, this register is cleared to ‘0’ by read access. When CSR0_bit0 is ‘0’, this register is cleared to ‘0’ by write access of any data. The condition of counting up is following. When the frame which has TCP/UDP/ICMP Checksum Error comes, TOE counts up this register.

30.4.3.10 Filter Operation/Auto Response Enable register (CSFR00)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	arp_ns_abt_en	—	—	reply_fifo_en	reg_fifo_en	true_false	ieee_len_fen	ana_stp_fen	v6_ana_protocol_fen	ns_dis_fen	ns_dup_fen	ns_uni_nopt_fen	ns_uni_fen
Initial Value	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_mul_fen	arp_noreq_fen	garp_self_fen	garp_other_fen	arp_self_fen	arp_other_fen	mac_da_mul_fen	mac_da_bro_fen	mac_da_uni_fen	v6_udp_pt_fen	v4_udp_pt_fen	v6_protocol_fen	v4_protocol_fen	type_fen	sby_mode[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	arp_ns_abt_en	1b	R/W	ARP ns frame Abort Enable Configuration of Transferred to DMAC or Discarded for the frame which was stored to REPLY FIFO of REG FIFO. 0b: Transferred to DMAC 1b: Discarded (Aborted)
27 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	reply_fifo_en	0b	R/W	Reply FIFO Enable Enable of Auto Response Function for ARP REQ and Neighbor Solicitations 0b: Disable 1b: Enable
24	reg_fifo_en	0b	R/W	Reg FIFO Enable Enable of Interrupt which occurs by reception of ARP REQ and Neighbor Solicitations. 0b: Disable 1b: Enable
23	true_false	0b	R/W	True False mode Discarding condition of the frame by filter. 0b: Discard by unmatch to filter condition 1b: Discard by match to filter condition
22	ieee_len_fen	1b	R/W	IEEE Length Filter Enable Filtering Operation for the IEEE802.3 Length Frame. 0b: Non-Filtering 1b: Filtering
21	ana_stp_fen	0b	R/W	Analysis stop Filter Enable Filtering Operation for the case of stopping frame analysis due to impossible analysis. 0b: Non-Filtering 1b: Filtering
20	v6_ana_protocol_fen	0b	R/W	IPv6 Analysis protocol Filter Enable Filtering Operation for the case of stopping frame analysis because Protocol No is not match to configurated value by v6_ana_protocol_0-15. 0b: Non-Filtering 1b: Filtering
19	ns_dis_fen	0b	R/W	Neighbor Solicitation discarded Filter Enable Filtering Operation for the invalid Neighbor Solicitations. 0b: Non-Filtering 1b: Filtering

Bit	Bit Name	Initial Value	R/W	Description
18	ns_dup_fen	0b	R/W	Neighbor Solicitation Duplicate Filter Enable Filtering Operation for the IP Duplicate Detection Neighbor Solicitations of Local Station. 0b: Non-Filtering 1b: Filtering
17	ns_uni_nopt_fen	0b	R/W	Neighbor Solicitation Unicast no option Filter Enable Filtering Operation for the Unicast Neighbor Solicitations of Local Station. (Except “Option Type=1” or No “Option Type”) 0b: Non-Filtering 1b: Filtering
16	ns_uni_fen	0b	R/W	Neighbor Solicitation Unicast Filter Enable Filtering Operation for the Unicast Neighbor Solicitations of Local Station. (Option Type=1) 0b: Non-Filtering 1b: Filtering
15	ns_mul_fen	0b	R/W	Neighbor Solicitation Multicast Filter Enable Filtering Operation for the Multicast Neighbor Solicitations of Local Station. 0b: Non-Filtering 1b: Filtering
14	arp_noreq_fen	0b	R/W	ARP No request Filer Enable Filtering Operation for the ARP frame except ARP request. 0b: Non-Filtering 1b: Filtering
13	garp_self_fen	0b	R/W	GARP Self Filter Enable Filtering Operation for the GARP request of Local Station. 0b: Non-Filtering 1b: Filtering
12	garp_other_fen	0b	R/W	GARP Other Filter Enable Filtering Operation for the GARP request of Other Station. 0b: Non-Filtering 1b: Filtering
11	arp_self_fen	0b	R/W	ARP Self Filter Enable Filtering Operation for the ARP request of Local Station. 0b: Non-Filtering 1b: Filtering
10	arp_other_fen	0b	R/W	ARP Other Filter Enable Filtering Operation for the ARP request of Other Station. 0b: Non-Filtering 1b: Filtering
9	mac_da_mul_fen	0b	R/W	MAC DA Multicast Filter Enable Filtering Operation by MAC DA. (multicast) 0b: Non-Filtering 1b: Filtering
8	mac_da_bro_fen	0b	R/W	MAC DA Broadcast Filter Enable Filtering Operation by MAC DA. (broadcast) 0b: Non-Filtering 1b: Filtering
7	mac_da_uni_fen	0b	R/W	MAC DA Unicast Filter Enable Filtering Operation by MAC DA. (unicast) 0b: Non-Filtering 1b: Filtering

Bit	Bit Name	Initial Value	R/W	Description
6	v6_udp_pt_fen	0b	R/W	IPv6 UDP Port Filter Enable Filtering Operation by IPv6 UDP port number. (destination) 0b: Non-Filtering 1b: Filtering
5	v4_udp_pt_fen	0b	R/W	IPv4 UDP Port Filter Enable Filtering Operation by IPv4 UDP port number. (destination) 0b: Non-Filtering 1b: Filtering
4	v6_protocol_fen	0b	R/W	IPv6 protocol Filter Enable Filtering Operation by IPv6 Protocol number. 0b: Non-Filtering 1b: Filtering
3	v4_protocol_fen	0b	R/W	IPv4 protocol Filter Enable Filtering Operation by IPv4 Protocol number. 0b: Non-Filtering 1b: Filtering
2	type_fen	0b	R/W	Type Filter Enable Filtering Operation by Ethernet TYPE/TPID. 0b: Non-Filtering 1b: Filtering
1 to 0	sby_mode	00b	R/W	Standby Mode Filtering Operation Mode 00b: Non-Filtering 1*b: Filtering

30.4.3.11 Local IPv4 address register (CSFR01)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ipv4_adr[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ipv4_adr[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ipv4_adr	H’0000_0000	R/W	IPv4 address IPv4 Address of Local Station Write to this field when CSFR00.reply_fifo_en = “0”, reg_fifo_en = “0” and sby_mode = “00b”.

30.4.3.12 Local IPv6 address register i (i = 0 to 3) (CSFR02_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ipv6_adr[32*j+31:32*j+16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ipv6_adr[32*j+15:32*j+0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ipv6_adr	H’0000_0000	R/W	IPv6 Address IPv6 Address[32*j+31:32*j] of Local Station The relationship between ‘i’ and ‘j’ is the following. i = 0 -> j = 3 i = 1 -> j = 2 i = 2 -> j = 1 i = 3 -> j = 0 Write to this field when CSFR00.reply_fifo_en = “0”, reg_fifo_en = “0” and sby_mode = “00b”.

30.4.3.13 Upper Local MAC address register (CSFR03_U)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_adr[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_adr	H’0000	R/W	MAC Address MAC Address [47:32] of Local Station Write to this field when CSFR00.reply_fifo_en = “0”, reg_fifo_en = “0” and sby_mode = “00b”.

30.4.3.14 Lower Local MAC address register (CSFR03_L)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	mac_adr[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_adr[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_adr	H’0000_0000	R/W	MAC Address MAC Address [32:0] of Local Station Write to this field when CSFR00.reply_fifo_en = “0”, reg_fifo_en = “0” and sby_mode = “00b”.

30.4.3.15 IPv6 no_next_header Protocol Number register (CSFR04)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	no_next_header[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	no_next_header	H'3B	R/W	Protocol Number of IPv6 no_next_header Usually, do not need to change from Initial values (59). Write to this field when CSFR00.sby mode is '00b'.

30.4.3.16 Ethernet Type Condition register i (i = 0 to 3) (CSFR10_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	type_2*i+1[15:8]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	type_2*I[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	type_2*i+1	H'0000	R/W	Ethernet Type 2*i+1 Configuration of the Ethernet TYPE for Filter Condition and Interrupt Condition. Write to this field when CSFR10. type_2*i+1_en is "0".
15 to 0	type_2*i	H'0000	R/W	Ethernet Type 2*i Configuration of the Ethernet TYPE for Filter Condition and Interrupt Condition. Write to this field when CSFR10. type_2*i_en is "0".

30.4.3.17 Ethernet Type Condition Enable register (CSFR10)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	type_7_en	type_6_en	type_5_en	type_4_en	type_3_en	type_2_en	type_1_en	type_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	type_7_en	0b	R/W	type_7 enable Enable for the value of type_7 which is configured in CSFR10_i. 0b: Disable 1b: Enable
6	type_6_en	0b	R/W	type_6 enable Enable for the value of type_6 which is configured in CSFR10_i. 0b: Disable 1b: Enable
5	type_5_en	0b	R/W	type_5 enable Enable for the value of type_5 which is configured in CSFR10_i. 0b: Disable 1b: Enable
4	type_4_en	0b	R/W	type_4 enable Enable for the value of type_4 which is configured in CSFR10_i. 0b: Disable 1b: Enable
3	type_3_en	0b	R/W	type_3 enable Enable for the value of type_3 which is configured in CSFR10_i. 0b: Disable 1b: Enable
2	type_2_en	0b	R/W	type_2 enable Enable for the value of type_2 which is configured in CSFR10_i. 0b: Disable 1b: Enable
1	type_1_en	0b	R/W	type_1 enable Enable for the value of type_1 which is configured in CSFR10_i. 0b: Disable 1b: Enable
0	type_0_en	0b	R/W	type_0 enable Enable for the value of type_0 which is configured in CSFR10_i. 0b: Disable 1b: Enable

30.4.3.18 Protocol Condition register i (i = 0 to 3) (CSFR11_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	protocol_4*i+3[7:0]								protocol_4*i+2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	protocol_4*i+1[7:0]								protocol_4*I[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	protocol_4*i+3	H’00	R/W	protocol No 4*i+3 Configuration of the Protocol No for Filter Condition and Interrupt Condition. Write to this field when CSFR11. protocol_4*i+3_en is “0”.
23 to 16	protocol_4*i+2	H’00	R/W	protocol No 4*i+2 Configuration of the Protocol No for Filter Condition and Interrupt Condition. Write to this field when CSFR11. protocol_4*i+2_en is “0”.
15 to 8	protocol_4*i+1	H’00	R/W	protocol No 4*i+1 Configuration of the Protocol No for Filter Condition and Interrupt Condition. Write to this field when CSFR11. protocol_4*i+1_en is “0”.
7 to 0	protocol_4*i	H’00	R/W	protocol No 4*i Write to this field when CSFR11. protocol_4*i_en is “0”.

30.4.3.19 Protocol Condition Enable register (CSFR11)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	protocol_15_en	protocol_14_en	protocol_13_en	protocol_12_en	protocol_11_en	protocol_10_en	protocol_9_en	protocol_8_en	protocol_7_en	protocol_6_en	protocol_5_en	protocol_4_en	protocol_3_en	protocol_2_en	protocol_1_en	protocol_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	protocol_15_en	0b	R/W	protocol_15 enable Enable for the value of protocol_15 which is configured in CSFR11_i. 0b: Disable 1b: Enable
14	protocol_14_en	0b	R/W	protocol_14 enable Enable for the value of protocol_14 which is configured in CSFR11_i. 0b: Disable 1b: Enable
13	protocol_13_en	0b	R/W	protocol_13 enable Enable for the value of protocol_13 which is configured in CSFR11_i. 0b: Disable 1b: Enable
12	protocol_12_en	0b	R/W	protocol_12 enable Enable for the value of protocol_12 which is configured in CSFR11_i. 0b: Disable 1b: Enable
11	protocol_11_en	0b	R/W	protocol_11 enable Enable for the value of protocol_11 which is configured in CSFR11_i. 0b: Disable 1b: Enable
10	protocol_10_en	0b	R/W	protocol_10 enable Enable for the value of protocol_10 which is configured in CSFR11_i. 0b: Disable 1b: Enable
9	protocol_9_en	0b	R/W	protocol_9 enable Enable for the value of protocol_9 which is configured in CSFR11_i. 0b: Disable 1b: Enable
8	protocol_8_en	0b	R/W	protocol_8 enable Enable for the value of protocol_8 which is configured in CSFR11_i. 0b: Disable 1b: Enable
7	protocol_7_en	0b	R/W	protocol_7 enable Enable for the value of protocol_7 which is configured in CSFR11_i. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
6	protocol_6_en	0b	R/W	protocol_6 enable Enable for the value of protocol_6 which is configured in CSFR11_i. 0b: Disable 1b: Enable
5	protocol_5_en	0b	R/W	protocol_5 enable Enable for the value of protocol_5 which is configured in CSFR11_i. 0b: Disable 1b: Enable
4	protocol_4_en	0b	R/W	protocol_4 enable Enable for the value of protocol_4 which is configured in CSFR11_i. 0b: Disable 1b: Enable
3	protocol_3_en	0b	R/W	protocol_3 enable Enable for the value of protocol_3 which is configured in CSFR11_i. 0b: Disable 1b: Enable
2	protocol_2_en	0b	R/W	protocol_2 enable Enable for the value of protocol_2 which is configured in CSFR11_i. 0b: Disable 1b: Enable
1	protocol_1_en	0b	R/W	protocol_1 enable Enable for the value of protocol_1 which is configured in CSFR11_i. 0b: Disable 1b: Enable
0	protocol_0_en	0b	R/W	protocol_0 enable Enable for the value of protocol_0 which is configured in CSFR11_i. 0b: Disable 1b: Enable

30.4.3.20 UDP Por Condition register i (i = 0 to 11) (CSFR12_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	udp_pt_2*i+1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	udp_pt_2*I[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	udp_pt_2*i+1	H'0000	R/W	UDP port No. Configuration of the UDP Port No for Filter Condition and Interrupt Condition. Write to this field when CSFR12. udp_pt_2*i+1_en is “0”.
15 to 0	udp_pt_2*i	H'0000	R/W	UDP port No. Configuration of the UDP Port No for Filter Condition and Interrupt Condition. Write to this field when CSFR12. udp_pt_2*i_en is “0”.

30.4.3.21 UDP Port Condition Enable register (CSFR12)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	udp_pt_23_en	udp_pt_22_en	udp_pt_21_en	udp_pt_20_en	udp_pt_19_en	udp_pt_18_en	udp_pt_17_en	udp_pt_16_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	udp_pt_15_en	udp_pt_14_en	udp_pt_13_en	udp_pt_12_en	udp_pt_11_en	udp_pt_10_en	udp_pt_9_en	udp_pt_8_en	udp_pt_7_en	udp_pt_6_en	udp_pt_5_en	udp_pt_4_en	udp_pt_3_en	udp_pt_2_en	udp_pt_1_en	udp_pt_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	udp_pt_23_en	0b	R/W	udp_pt_23 enable Enable for the value of udp_pt_23 which is configured in CSFR12_i. 0b: Disable 1b: Enable
22	udp_pt_22_en	0b	R/W	udp_pt_22 enable Enable for the value of udp_pt_22 which is configured in CSFR12_i. 0b: Disable 1b: Enable
21	udp_pt_21_en	0b	R/W	udp_pt_21 enable Enable for the value of udp_pt_21 which is configured in CSFR12_i. 0b: Disable 1b: Enable
20	udp_pt_20_en	0b	R/W	udp_pt_20 enable Enable for the value of udp_pt_20 which is configured in CSFR12_i. 0b: Disable 1b: Enable
19	udp_pt_19_en	0b	R/W	udp_pt_19 enable Enable for the value of udp_pt_19 which is configured in CSFR12_i. 0b: Disable 1b: Enable
18	udp_pt_18_en	0b	R/W	udp_pt_18 enable Enable for the value of udp_pt_18 which is configured in CSFR12_i. 0b: Disable 1b: Enable
17	udp_pt_17_en	0b	R/W	udp_pt_17 enable Enable for the value of udp_pt_17 which is configured in CSFR12_i. 0b: Disable 1b: Enable
16	udp_pt_16_en	0b	R/W	udp_pt_16 enable Enable for the value of udp_pt_16 which is configured in CSFR12_i. 0b: Disable 1b: Enable
15	udp_pt_15_en	0b	R/W	udp_pt_15 enable Enable for the value of udp_pt_15 which is configured in CSFR12_i. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
14	udp_pt_14_en	0b	R/W	udp_pt_14 enable Enable for the value of udp_pt_14 which is configured in CSFR12_i. 0b: Disable 1b: Enable
13	udp_pt_13_en	0b	R/W	udp_pt_13 enable Enable for the value of udp_pt_13 which is configured in CSFR12_i. 0b: Disable 1b: Enable
12	udp_pt_12_en	0b	R/W	udp_pt_12 enable Enable for the value of udp_pt_12 which is configured in CSFR12_i. 0b: Disable 1b: Enable
11	udp_pt_11_en	0b	R/W	udp_pt_11 enable Enable for the value of udp_pt_11 which is configured in CSFR12_i. 0b: Disable 1b: Enable
10	udp_pt_10_en	0b	R/W	udp_pt_10 enable Enable for the value of udp_pt_10 which is configured in CSFR12_i. 0b: Disable 1b: Enable
9	udp_pt_9_en	0b	R/W	udp_pt_9 enable Enable for the value of udp_pt_9 which is configured in CSFR12_i. 0b: Disable 1b: Enable
8	udp_pt_8_en	0b	R/W	udp_pt_8 enable Enable for the value of udp_pt_8 which is configured in CSFR12_i. 0b: Disable 1b: Enable
7	udp_pt_7_en	0b	R/W	udp_pt_7 enable Enable for the value of udp_pt_7 which is configured in CSFR12_i. 0b: Disable 1b: Enable
6	udp_pt_6_en	0b	R/W	udp_pt_6 enable Enable for the value of udp_pt_6 which is configured in CSFR12_i. 0b: Disable 1b: Enable
5	udp_pt_5_en	0b	R/W	udp_pt_5 enable Enable for the value of udp_pt_5 which is configured in CSFR12_i. 0b: Disable 1b: Enable
4	udp_pt_4_en	0b	R/W	udp_pt_4 enable Enable for the value of udp_pt_4 which is configured in CSFR12_i. 0b: Disable 1b: Enable
3	udp_pt_3_en	0b	R/W	udp_pt_3 enable Enable for the value of udp_pt_3 which is configured in CSFR12_i. 0b: Disable 1b: Enable
2	udp_pt_2_en	0b	R/W	udp_pt_2 enable Enable for the value of udp_pt_2 which is configured in CSFR12_i. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
1	udp_pt_1_en	0b	R/W	udp_pt_1 enable Enable for the value of udp_pt_1 which is configured in CSFR12_i. 0b: Disable 1b: Enable
0	udp_pt_0_en	0b	R/W	udp_pt_0 enable Enable for the value of udp_pt_0 which is configured in CSFR12_i. 0b: Disable 1b: Enable

30.4.3.22 Upper MAC DA unicast address Condition register (CSFR13_U)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_uni[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_da_uni	H’0000	R/W	MAC DA unicast address Configuration of the MAC DA unicast Address [47:32] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_uni_en = “0”.

30.4.3.23 Lower MAC DA unicast address Condition register (CSFR13_L)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	mac_da_uni[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_uni[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_da_uni	H’0000_0000	R/W	MAC DA Unicast address Configuration of the MAC DA unicast Address [31:0] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_uni_en = “0”.

30.4.3.24 Upper MAC DA broadcast address Condition register (CSFR14_U)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_bro[47:40]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_da_bro	H'FFFF	R/W	MAC DA Broadcast address Configuration of the MAC DA broadcast Address [47:32] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_bro_en = "0".

30.4.3.25 Lower MAC DA broadcast address Condition register (CSFR14_L)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	mac_da_bro[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_bro[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_da_bro	H'FFFF_FFFF	R/W	MAC DA Broadcast address Configuration of the MAC DA broadcast Address [31:0] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_bro_en = "0".

30.4.3.26 Upper MAC DA multicast address Condition register i (i = 0 to 19) (CSFR15_U_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_mul_i[47:32]															
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	mac_da_mul_i	H'0100	R/W	MAC DA Multicast address Configuration of the MAC DA multicast Address [47:32] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_mul_i_en = "0".

30.4.3.27 Lower MAC DA multicast address Condition register i (i = 0 to 19) (CSFR15_L_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	mac_da_mul_i[31:16]															
Initial Value	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_mul_i[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	mac_da_mul_i	H'5E00_0000	R/W	MAC DA Multicast address Configuration of the MAC DA multicast Address [31:0] for Filter Condition and Interrupt Condition. Write to this field when CSFR15.mac_da_mul_i _en = "0".

30.4.3.28 MAC DA Condition Enable register (CSFR15)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	mac_da_uni_en	—	—	—	mac_da_bro_en	—	—	—	—	mac_da_mul_19_en	mac_da_mul_18_en	mac_da_mul_17_en	mac_da_mul_16_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	mac_da_mul_15_en	mac_da_mul_14_en	mac_da_mul_13_en	mac_da_mul_12_en	mac_da_mul_11_en	mac_da_mul_10_en	mac_da_mul_9_en	mac_da_mul_8_en	mac_da_mul_7_en	mac_da_mul_6_en	mac_da_mul_5_en	mac_da_mul_4_en	mac_da_mul_3_en	mac_da_mul_2_en	mac_da_mul_1_en	mac_da_mul_0_en
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	mac_da_uni_en	0b	R/W	Mac_da_uni enable Enable for the value of mac_da_uni which is configured in CSFR13_U/L. 0b: Disable 1b: Enable
27 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	mac_da_bro_en	0b	R/W	Mac_da_bro enable Enable for the value of mac_da_bro which is configured in CSFR14_U/L. 0b: Disable 1b: Enable
23 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	mac_da_mul_19_en	0b	R/W	mac_da_mul_19 enable Enable for the value of mac_da_mul_19 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
18	mac_da_mul_18_en	0b	R/W	mac_da_mul_18 enable Enable for the value of mac_da_mul_18 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
17	mac_da_mul_17_en	0b	R/W	mac_da_mul_17 enable Enable for the value of mac_da_mul_17 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
16	mac_da_mul_16_en	0b	R/W	mac_da_mul_16 enable Enable for the value of mac_da_mul_16 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
15	mac_da_mul_15_en	0b	R/W	mac_da_mul_15 enable Enable for the value of mac_da_mul_15 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
14	mac_da_mul_14_en	0b	R/W	mac_da_mul_14 enable Enable for the value of mac_da_mul_14 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
13	mac_da_mul_13_en	0b	R/W	mac_da_mul_13 enable Enable for the value of mac_da_mul_13 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
12	mac_da_mul_12_en	0b	R/W	mac_da_mul_12 enable Enable for the value of mac_da_mul_12 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
11	mac_da_mul_11_en	0b	R/W	mac_da_mul_11 enable Enable for the value of mac_da_mul_11 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
10	mac_da_mul_10_en	0b	R/W	mac_da_mul_10 enable Enable for the value of mac_da_mul_10 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
9	mac_da_mul_9_en	0b	R/W	mac_da_mul_9 enable Enable for the value of mac_da_mul_9 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
8	mac_da_mul_8_en	0b	R/W	mac_da_mul_8 enable Enable for the value of mac_da_mul_8 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
7	mac_da_mul_7_en	0b	R/W	mac_da_mul_7 enable Enable for the value of mac_da_mul_7 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
6	mac_da_mul_6_en	0b	R/W	mac_da_mul_6 enable Enable for the value of mac_da_mul_6 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
5	mac_da_mul_5_en	0b	R/W	mac_da_mul_5 enable Enable for the value of mac_da_mul_5 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
4	mac_da_mul_4_en	0b	R/W	mac_da_mul_4 enable Enable for the value of mac_da_mul_4 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
3	mac_da_mul_3_en	0b	R/W	mac_da_mul_3 enable Enable for the value of mac_da_mul_3 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
2	mac_da_mul_2_en	0b	R/W	mac_da_mul_2 enable Enable for the value of mac_da_mul_2 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
1	mac_da_mul_1_en	0b	R/W	mac_da_mul_1 enable Enable for the value of mac_da_mul_1 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable
0	mac_da_mul_0_en	0b	R/W	mac_da_mul_0 enable Enable for the value of mac_da_mul_0 which is configured in CSFR15_U/L. 0b: Disable 1b: Enable

30.4.3.29 IPv6 analysis Protocol Condition register 0 (CSFR16_0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	v6_ana_protocol_3[7:0]								v6_ana_protocol_2[7:0]							
Initial Value	0	0	1	0	1	0	0	1	0	0	0	1	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	v6_ana_protocol_1[7:0]								v6_ana_protocol_0[7:0]							
Initial Value	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	v6_ana_protocol_3	H'29	R/W	IPv6 Analyzable Protocol No 3 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'29 IPV6 (41). Write to this field when CSFR16. v6_ana_protocol_3_en is '0b'.
23 to 16	v6_ana_protocol_2	H'11	R/W	IPv6 Analyzable Protocol No 2 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'11 UDP (17). Write to this field when CSFR16. v6_ana_protocol_2_en is '0b'.
15 to 8	v6_ana_protocol_1	H'06	R/W	IPv6 Analyzable Protocol No 1 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'06 TCP (6). Write to this field when CSFR16. v6_ana_protocol_1_en is '0b'.
7 to 0	v6_ana_protocol_0	H'00	R/W	IPv6 Analyzable Protocol No 0 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'00 Hop-by-Hop (0). Write to this field when CSFR16. v6_ana_protocol_0_en is '0b'.

30.4.3.30 IPv6 analysis Protocol Condition register 1 (CSFR16_1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	v6_ana_protocol_7[7:0]								v6_ana_protocol_6[7:0]							
Initial Value	0	0	1	1	1	0	1	0	0	0	1	1	0	0	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	v6_ana_protocol_5[7:0]								v6_ana_protocol_4[7:0]							
Initial Value	0	0	1	0	1	1	0	0	0	0	1	0	1	0	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	v6_ana_protocol_7	H'3A	R/W	IPv6 Analyzable Protocol No.7 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'3A ICMPv6 (58). Write to this field when CSFR16. v6_ana_protocol_7_en is '0b'.
23 to 16	v6_ana_protocol_6	H'33	R/W	IPv6 Analyzable Protocol No.6 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'33 Authentication (51). Write to this field when CSFR16. v6_ana_protocol_6_en is '0b'.
15 to 8	v6_ana_protocol_5	H'2C	R/W	IPv6 Analyzable Protocol No.5 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'2C Flag (44). Write to this field when CSFR16. v6_ana_protocol_5_en is '0b'.
7 to 0	v6_ana_protocol_4	H'2B	R/W	IPv6 Analyzable Protocol No.4 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H'2B Routing (43). Write to this field when CSFR16. v6_ana_protocol_4_en is '0b'.

30.4.3.31 IPv6 analysis Protocol Condition register 2 (CSFR16_2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	v6_ana_protocol_8[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	v6_ana_protocol_8	H’3C	R/W	IPv6 Analyzable Protocol No 8 IPv6 protocol No to continue analysis in the Frame analysis procedure of Filter and Interrupt. Initial value is H’3C Destination Options (60). Write to this field when CSFR16. v6 ana protocol 8 en is ‘0b’.

30.4.3.32 IPv6 analysis Protocol Condition Enable register (CSFR16)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	v6_ana_protocol_8_en	v6_ana_protocol_7_en	v6_ana_protocol_6_en	v6_ana_protocol_5_en	v6_ana_protocol_4_en	v6_ana_protocol_3_en	v6_ana_protocol_2_en	v6_ana_protocol_1_en	v6_ana_protocol_0_en
Initial Value	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	v6_ana_protocol_8_en	1b	R/W	v6_ana_protocol_8 enable Enable for the value of v6_ana_protocol_8 which is configured in CSFR16_i. 0b: Disable 1b: Enable
7	v6_ana_protocol_7_en	1b	R/W	v6_ana_protocol_7 enable Enable for the value of v6_ana_protocol_7 which is configured in CSFR16_i. 0b: Disable 1b: Enable
6	v6_ana_protocol_6_en	1b	R/W	v6_ana_protocol_6 enable Enable for the value of v6_ana_protocol_6 which is configured in CSFR16_i. 0b: Disable 1b: Enable
5	v6_ana_protocol_5_en	1b	R/W	v6_ana_protocol_5 enable Enable for the value of v6_ana_protocol_5 which is configured in CSFR16_i. 0b: Disable 1b: Enable
4	v6_ana_protocol_4_en	1b	R/W	v6_ana_protocol_4 enable Enable for the value of v6_ana_protocol_4 which is configured in CSFR16_i. 0b: Disable 1b: Enable
3	v6_ana_protocol_3_en	1b	R/W	v6_ana_protocol_3 enable Enable for the value of v6_ana_protocol_3 which is configured in CSFR16_i. 0b: Disable 1b: Enable
2	v6_ana_protocol_2_en	1b	R/W	v6_ana_protocol_2 enable Enable for the value of v6_ana_protocol_2 which is configured in CSFR16_i. 0b: Disable 1b: Enable
1	v6_ana_protocol_1_en	1b	R/W	v6_ana_protocol_1 enable Enable for the value of v6_ana_protocol_1 which is configured in CSFR16_i. 0b: Disable 1b: Enable
0	v6_ana_protocol_0_en	1b	R/W	v6_ana_protocol_0 enable Enable for the value of v6_ana_protocol_0 which is configured in CSFR16_i. 0b: Disable 1b: Enable

30.4.3.33 Wake Up Interrupt Status register (CSFR20)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	trs_dmac	—	—	—	—	—	—	—	ieee_len	ns_dis	ns_dup	ns_uni_nopt	ns_uni
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_mul	ana_stp	v6_ana_protocol	arp_noreq	garp_self	garp_other	arp_self	arp_other	v6_udp_pt	v4_udp_pt	v6_protocol	v4_protocol	type	mac_da_mul	mac_da_bro	mac_da_uni
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	trs_dmac	0b	R/W	Transmit to DMAC Wake Up Interrupt which occurs by transmitting Frame from TOE to DMAC without discarding of filtering ‘1’ Write: Clear ‘0’ Write: Invalid
27 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	ieee_len	0b	R/W	IEEE Length filter Wake Up Interrupt which occurs by reception of IEEE802.3 Length Frame ‘1’ Write: Clear ‘0’ Write: Invalid
19	ns_dis	0b	R/W	Neighbor Solicitation discarded filter Wake Up Interrupt which occurs by reception of Invalid Neighbor Solicitations ‘1’ Write: Clear ‘0’ Write: Invalid
18	ns_dup	0b	R/W	Neighbor Solicitation Duplicate filter Wake Up Interrupt which occurs by reception of IP Duplicate Detection Neighbor Solicitations. ‘1’ Write: Clear ‘0’ Write: Invalid
17	ns_uni_nopt	0b	R/W	Neighbor Solicitation Unicast no option filter Wake Up Interrupt which occurs by reception of unicast Neighbor Solicitations (Except “Option Type=1” or No “Option Type”). ‘1’ Write: Clear ‘0’ Write: Invalid
16	ns_uni	0b	R/W	Neighbor Solicitation Unicast filter Wake Up Interrupt which occurs by reception of unicast Neighbor Solicitations (“Option Type=1”). ‘1’ Write: Clear ‘0’ Write: Invalid
15	ns_mul	0b	R/W	Neighbor Solicitation Multicast filter Wake Up Interrupt which occurs by reception of multicast Neighbor Solicitations. ‘1’ Write: Clear ‘0’ Write: Invalid

Bit	Bit Name	Initial Value	R/W	Description
14	ana_stp	0b	R/W	Analysis stop filter Wake Up Interrupt which occurs by reception of Un-analyzable Frames. ‘1’ Write: Clear ‘0’ Write: Invalid
13	v6_ana_protocol	0b	R/W	IPv6 Analysis protocol filter Wake Up Interrupt which occurs by reception of Un-analyzable IPv6 Protocol No. ‘1’ Write: Clear ‘0’ Write: Invalid
12	arp_noreq	0b	R/W	ARP no request filer Wake Up Interrupt which occurs by reception of ARP (Except ARP REQ). ‘1’ Write: Clear ‘0’ Write: Invalid
11	garp_self	0b	R/W	GARP self-filter Wake Up Interrupt which occurs by reception of GARP REQ for Local Station. ‘1’ Write: Clear ‘0’ Write: Invalid
10	garp_other	0b	R/W	GARP other filter Wake Up Interrupt which occurs by reception of GARP REQ for Other Station. ‘1’ Write: Clear ‘0’ Write: Invalid
9	arp_self	0b	R/W	ARP self-filter Wake Up Interrupt which occurs by reception of ARP REQ for Local Station. ‘1’ Write: Clear ‘0’ Write: Invalid
8	arp_other	0b	R/W	ARP other filter Wake Up Interrupt which occurs by reception of ARP REQ for Other Station. ‘1’ Write: Clear ‘0’ Write: Invalid
7	v6_udp_pt	0b	R/W	IPv6 UDP port filter Wake Up Interrupt which occurs by match of IPv6 UDP Port No. ‘1’ Write: Clear ‘0’ Write: Invalid
6	v4_udp_pt	0b	R/W	IPv4 UDP port filter Wake Up Interrupt which occurs by match of IPv4 UDP Port No. ‘1’ Write: Clear ‘0’ Write: Invalid
5	v6_protocol	0b	R/W	IPv6 protocol filter Wake Up Interrupt which occurs by match of IPv6 Protocol No. ‘1’ Write: Clear ‘0’ Write: Invalid
4	v4_protocol	0b	R/W	IPv4 protocol filter Wake Up Interrupt which occurs by match of IPv4 Protocol No. ‘1’ Write: Clear ‘0’ Write: Invalid
3	type	0b	R/W	Type filter Wake Up Interrupt which occurs by match of Ether Type. ‘1’ Write: Clear ‘0’ Write: Invalid
2	mac_da_mul	0b	R/W	MAC DA multicast filter Wake Up Interrupt which occurs by match of Multicast MAC DA. ‘1’ Write: Clear ‘0’ Write: Invalid

Bit	Bit Name	Initial Value	R/W	Description
1	mac_da_bro	0b	R/W	MAC DA broadcast filter Wake Up Interrupt which occurs by match of Broadcast MAC DA. ‘1’ Write: Clear ‘0’ Write: Invalid
0	mac_da_uni	0b	R/W	MAC DA unicast filter Wake Up Interrupt which occurs by match of Unicast MAC DA. ‘1’ Write: Clear ‘0’ Write: Invalid

NOTE

- From b[0] to b[20] are set to “1” when TOE received the Rx Frame which matches to Filter Conditions even if that frame has any error (such as CRC error, short length error or etc.). Therefore, use these bits only for Debug.
- b [28] is set to “1” only when TOE sent a normal Rx Frame to DMAC.

30.4.3.34 Wake Up Interrupt Mask register (CSFR21)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	trs_dmac_m	—	—	—	—	—	—	—	ieee_len_m	ns_dis_m	ns_dup_m	ns_uni_nopt_m	ns_uni_m
Initial Value	0	0	0	1	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_mul_m	ana_stp_m	v6_ana_protocol_m	arp_noreq_m	garp_self_m	garp_other_m	arp_self_m	arp_other_m	v6_udppt_m	v4_udppt_m	v6_protocol_m	v4_protocol_m	type_m	mac_da_mul_m	mac_da_bro_m	mac_da_uni_m
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	trs_dmac_m	1b	R/W	Transmit to DMAC Mask Mask Configuration for Wake-Up Interrupt which occurs by transmitting Frame from TOE to DMAC without discarding of filtering 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
27 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	ieee_len_m	1b	R/W	IEEE Length filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of IEEE802.3 Length Frame 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
19	ns_dis_m	1b	R/W	Neighbor Solicitation discarded filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of Invalid Neighbor Solicitations 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
18	ns_dup_m	1b	R/W	Neighbor Solicitation Duplicate filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of IP Duplicate Detection Neighbor Solicitations. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
17	ns_uni_nopt_m	1b	R/W	Neighbor Solicitation Unicast no option filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of unicast Neighbor Solicitations (Except “Option Type=1” or No “Option Type”). 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
16	ns_uni_m	1b	R/W	Neighbor Solicitation Unicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of unicast Neighbor Solicitations (“Option Type=1”). 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
15	ns_mul_m	1b	R/W	Neighbor Solicitation Multicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of multicast Neighbor Solicitations. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

Bit	Bit Name	Initial Value	R/W	Description
14	ana_stp_m	1b	R/W	Analysis stop filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of Un-analyzable Frames. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
13	v6_ana_protocol_m	1b	R/W	IPv6 Analysis protocol filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of Un-analyzable IPv6 Protocol No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
12	arp_noreq_m	1b	R/W	ARP no request filer Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of ARP (Except ARP REQ). 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
11	garp_self_m	1b	R/W	GARP self-filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of GARP REQ for Local Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
10	garp_other_m	1b	R/W	GARP other filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of GARP REQ for Other Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
9	arp_self_m	1b	R/W	ARP self-filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of ARP REQ for Local Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
8	arp_other_m	1b	R/W	ARP other filter Mask Mask Configuration for Wake-Up Interrupt which occurs by reception of ARP REQ for Other Station. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
7	v6_udp_pt_m	1b	R/W	IPv6 UDP port filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv6 UDP Port No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
6	v4_udp_pt_m	1b	R/W	IPv4 UDP port filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv4 UDP Port No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
5	v6_protocol_m	1b	R/W	IPv6 protocol filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv6 Protocol No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
4	v4_protocol_m	1b	R/W	IPv4 protocol filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of IPv4 Protocol No. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

Bit	Bit Name	Initial Value	R/W	Description
3	type_m	1b	R/W	Type filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Ether Type. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
2	mac_da_mul_m	1b	R/W	MAC DA multicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Multicast MAC DA. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
1	mac_da_bro_m	1b	R/W	MAC DA broadcast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Broadcast MAC DA. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
0	mac_da_uni_m	1b	R/W	MAC DA unicast filter Mask Mask Configuration for Wake-Up Interrupt which occurs by match of Unicast MAC DA. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

30.4.3.35 Auto Response Configuration register1 (CSFR30)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	na_dup_of
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ns_dup_int_en	ns_uni_nopt_int_en	ns_uni_int_en	ns_mul_int_en	ns_dup_reply_en	—	ns_uni_reply_en	ns_mul_reply_en	—	garp_self_int_en	garp_other_int_en	arp_self_int_en	—	garp_self_reply_en	—	arp_self_reply_en
Initial Value	0	1	1	1	1	0	1	1	0	0	1	1	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	na_dup_of	1b	R/W	Neighbor Advertisement Duplicate Override flag Configuration of Override flag to Auto Response Neighbor Advertisements when the reception of IP Duplicate Detection Neighbor Solicitation for Local Station. Usually, do not change this bit. Keep initial values “1b”. 0b: Override flag is set as 0 1b: Override flag is set as 1
15	ns_dup_int_en	0b	R/W	Neighbor Solicitation Duplicate interrupt enable Enable of Interrupt which occurs by reception of IP Duplicate Detection Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
14	ns_uni_nopt_int_en	1b	R/W	Neighbor Solicitation Unicast no option interrupt enable Enable of Interrupt which occurs by reception of Unicast Neighbor Solicitations for Local Station (Except “Option Type=1” or No “Option Type”). 0b: Disable 1b: Enable
13	ns_uni_int_en	1b	R/W	Neighbor Solicitation Unicast interrupt enable Enable of Interrupt which occurs by reception of Unicast Neighbor Solicitations for Local Station (Option Type=1). 0b: Disable 1b: Enable
12	ns_mul_int_en	1b	R/W	Neighbor Solicitation Multicast interrupt enable Enable of Interrupt which occurs by reception of Multicast Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
11	ns_dup_reply_en	1b	R/W	Neighbor Solicitation Duplicate reply enable Enable of Auto Response for reception of IP Duplicate Detection Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	ns_uni_reply_en	1b	R/W	Neighbor Solicitation Unicast reply enable Enable of Auto Response for reception of Unicast Neighbor Solicitations (Option Type=1) for Local Station. 0b: Disable 1b: Enable
8	ns_mul_reply_en	1b	R/W	Neighbor Solicitation Multicast reply enable Enable of Auto Response for reception of Multicast Neighbor Solicitations for Local Station. 0b: Disable 1b: Enable
7	—	0b	R	Reserved T Whenever it is read, 0 is read. The written value will be ignored.
6	garp_self_int_en	0b	R/W	GARP self-interrupt enable Enable of Interrupt which occurs by reception of GARP REQ for Local Station. 0b: Disable 1b: Enable
5	garp_other_int_en	1b	R/W	GARP other interrupt enable Enable of Interrupt which occurs by reception of GARP REQ for Other Station. 0b: Disable 1b: Enable
4	arp_self_int_en	1b	R/W	ARP self-interrupt enable Enable of Interrupt which occurs by reception of ARP REQ for Local Station. 0b: Disable 1b: Enable
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	garp_self_reply_en	1b	R/W	GARP self reply enable Enable of Auto Response for reception of GARP REQ for Local Station. 0b: Disable 1b: Enable
1	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	arp_self_reply_en	1b	R/W	ARP self reply enable Enable of Auto Response for reception of ARP REQ for Local Station. 0b: Disable 1b: Enable

30.4.3.36 Auto Response Configuration register2 (CSFR31)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	na_tc[7:0]								—	—	—	—	na_fl[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	na_fl[15:0]															
Initial Value	1	1	1	1	1	0	1	1	0	1	1	1	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	na_tc	H'00	R/W	Neighbor Advertisement Traffic Class Traffic Class of Auto Response Neighbor Advertisements for Received Neighbor Solicitations for Local Station. Write to this field when CSFR00.reply_fifo_en is “0”.
23 to 20	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19 to 0	na_fl	H'00000	R/W	Neighbor Advertisement Flow Label Flow Label of Auto Response Neighbor Advertisements for Received Neighbor Solicitations for Local Station. Write to this field when CSFR00.reply_fifo_en is “0”.

30.4.3.37 ARPREQ/Neighbor Solicitations Receive Interrupt Status register (CSFR40)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	reply_fifo_full	reg_fifo_full	arp_ns
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	reply_fifo_full	0b	R/W	Reply FIFO full This bit indicates that although TOE received ARPREQ or Neighbor Solicitations, but it was discarded because REPLY_FIFO is full. ‘1’ Write: Clear ‘0’ Write: Invalid
1	reg_fifo_full	0b	R/W	Reg FIFO full This bit indicates that although TOE received ARPREQ or Neighbor Solicitations, but it was discarded because REG_FIFO is full. ‘1’ Write: Clear ‘0’ Write: Invalid
0	arp_ns	0b	R/W	ARP Neighbor Solicitation This bit indicates that TOE received ARPREQ or Neighbor Solicitations and they are stored to REG_FIFO completely. ‘1’ Write: Clear ‘0’ Write: Invalid

30.4.3.38 ARPREQ/Neighbor Solicitations Receive Interrupt Mask register (CSFR41)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	reply_fifo_full_m	reg_fifo_full_m	arp_ns_m
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	reply_fifo_full_m	0b	R/W	Reply FIFO full Mask Mask Configuration for ARPREQ/Neighbor Solicitations Receive Interrupt which occurs by reply_fifo_full. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
1	reg_fifo_full_m	0b	R/W	Reg FIFO full Mask Mask Configuration for ARPREQ/Neighbor Solicitations Receive Interrupt which occurs by reg_fifo_full. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)
0	arp_ns_m	0b	R/W	ARP Neighbor Solicitation Mask Mask Configuration for ARPREQ/Neighbor Solicitations Receive Interrupt which occurs by arp_ns. 0b: Non-Mask (Interrupt is asserted) 1b: Mask (Interrupt is not asserted)

30.4.3.39 Rx ARPREQ/Neighbor Solicitation Upper MAC SA register (CSFR42_U)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ns_option_type[7:0]								—	—	—	—	—	—	arp_ns_spec	rx_arp_ns
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	arp_sa_mac[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	ns_option_type	H'00	R	Neighbor Solicitations option type This field indicates Received Neighbor Solicitations Option Type. "H'00" means that Received Frame did not have Option. When rx_arp_ns = 0 (ARPREQ), this field is invalid.
23 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	arp_ns_spec	0b	R	ARP Neighbor Solicitations spec This field indicates Received ARPREQ/Neighbor Solicitations Frame Type. The Case of ARPREQ (rx_arp_ns = 0) 0b: ARPREQ 1b: GARPREQ The Case of Neighbor Solicitations (rx_arp_ns = 1) 0b: Multicast Neighbor Solicitations 1b: Unicast Neighbor Solicitations
16	rx_arp_ns	0b	R	ARP Neighbor Solicitations This field indicates which frame did TOE receive ARPREQ or Neighbor Solicitations. 0b: ARPREQ 1b: Neighbor Solicitations
15 to 0	arp_sa_mac	H'0000	R	ARP source Address MAC This field indicates the Source MAC Address [47:32] of Received ARPREQ/Neighbor Solicitation. When rx_arp_ns = 1 and ns_option_type = H'00, this field is invalid.

[Update timing of this register]

This register is updated when REG FIFO is empty and the interrupt of int_arp_ns_n caused by "Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO" is asserted.

This register is updated when REG FIFO is not empty and the interrupt of int_arp_ns_n caused by "Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO" is cleared by CSFR40.arp_ns.

30.4.3.40 Rx ARPREQ/Neighbor Solicitation Lower MAC SA register (CSFR42_L)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	arp_sa_mac[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	arp_sa_mac[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	arp_sa_mac	H'0000_0000	R	ARP source Address MAC This field indicates the Source MAC Address [31:0] of Received ARPREQ/Neighbor Solicitation. When rx_arp_ns = 1 and ns_option_type = H'00, this field is invalid.

[Update timing of this register]

This register is updated when REG FIFO is empty and the interrupt of int_arp_ns_n caused by “Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO” is asserted.

This register is updated when REG FIFO is not empty and the interrupt of int_arp_ns_n caused by “Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO” is cleared by CSFR40.arp_ns.

30.4.3.41 Rx ARPREQ/Neighbor Solicitation IP SA register i (i = 0 to 3) (CSFR43_i)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	iarp_sa_ip[32*j+31:32*j+16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	arp_sa_ip[32*j+15:32*j+0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	arp_sa_ip	All 0	R	ARP source Address IP This field indicates the Source IP Address[32*j+31:32*j] of Received ARPREQ/Neighbor Solicitations. The relationship between ‘i’ and ‘j’ is the following. i = 0 -> j = 3 i = 1 -> j = 2 i = 2 -> j = 1 i = 3 -> j = 0 When rx_arp_ns = 0 (ARPREQ), Only Lower 32bit (CSFR43_3) is valid.

[Update timing of this register]

This register is updated when REG FIFO is empty and the interrupt of int_arp_ns_n caused by “Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO” is asserted.

This register is updated when REG FIFO is not empty and the interrupt of int_arp_ns_n caused by “Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO” is cleared by CSFR40.arp_ns.

30.5 Operation

The Gigabit Ethernet Interface consists of the following functional units:

- DMA transfer controller (DMAC):
  Handles DMA transfer between the data storage areas for reception and transmission in the URAM and the reception and transmission FIFO
- TCP/IP Offload Engine (TOE):
  Calculates Checksum by hardware. And also filters Ethernet Frames and responds specific frames automatically.
- MAC controller (E-MAC):
  Handles reception/transmission transfer from/to the RGMII or MII.

Using its direct memory access (DMA) function, the DMAC handles DMA transfer of frame data between the destinations for storing Ethernet frame data for transmission and reception in the URAM and the FIFO for reception and transmission. Data cannot be directly read from or written to the FIFO.

To handle DMA transfer, the DMAC requires information that includes the addresses for storage of data for transmission and received data. These data are referred to as descriptors. The DMAC reads data for transmission from the storage area for data to be transmitted according to the information in descriptors and writes received data to the storage area for received data accompanied by information in descriptors. The descriptors are placed in the URAM. Arranging multiple descriptors in descriptor lists allows the continuous reception or transmission of multiple Ethernet frames.

TOE calculates checksum of received frames from E-MAC. And then it outputted to DMAC. TOE also calculates checksum of transmission frames from DMAC. And then it outputted to E-MAC. When TOE received any frames from E-MAC, it analyzes them. By the configuration of registers, TOE filters received frames. If received frame does not correspond to the configured filter conditions by registers, it will be discarded. By using this analysis function, TOE is also able to respond to ARP Request and Neighbor Solicitation automatically.

The E-MAC supports a RGMII and MII, which provides an interface format for the externally connected PHY-LSI. The E-MAC transmits frames from TOE to the RGMII or MII. It also transmits frames from the RGMII or MII to TOE.

30.5.1 DMAC Operating Modes

Figure 30.2 illustrates the operating modes of the DMAC.

Transitions of DMAC operating mode are under the control of the items listed below.

- CPU operating mode (hardware reset)
- Configuration of the operating mode configuration bits (CCC.OPC) in the DMAC mode register

The operating mode can be confirmed by reading the operating mode status bits in the DMAC status register (CSR.OPS).

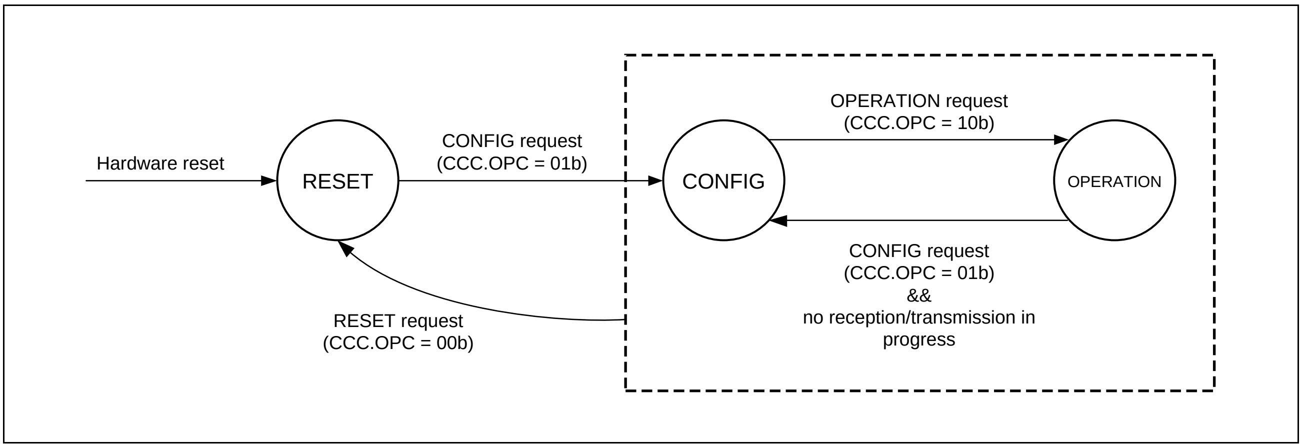


Figure 30.2 Operating Mode of DMAC

30.5.1.1 Operating Modes

(1) RESET mode

After a hardware reset, the DMAC enters RESET mode.

In RESET mode, only the DMAC operating mode control function is controllable and other functions are all stopped. This mode is designed for reduced power when the Ethernet function is not necessary.

(2) CONFIG mode

In CONFIG mode, various settings for the DMAC can be made.

The operation of most functions is stopped, and all status registers are initialized to their reset values.

(3) OPERATION mode

In OPERATION mode, all functions of the DMAC can operate. Ethernet communications can only proceed in this mode.

30.5.1.2 How to Set the Operating Mode

Set the operating mode configuration bits in the DMAC mode register (CCC.OPC) to select the operating mode. Furthermore, the operating mode can be checked by reading the operating mode status bits in the DMAC status register (CSR.OPS).

Transitions other than leaving from OPERATION mode are made after the value is written to the operating mode configuration bits (CCC.OPC).

For transitions leaving from OPERATION mode, it is not executed immediately because any transmission and reception in progress will be executed before leaving from OPERATION mode. To prevent any trouble, follow the “Stop Procedure” of **Section 30.5.13.4**.

30.5.1.3 Operating Mode Transitions Due to Hardware

The following hardware factors can also initiate transitions of the DMAC operating mode.

(1) Hardware reset

Resetting of the LSI chip leads to resetting of the entire Gigabit Ethernet Interface. The operating mode shifts to RESET mode.

30.5.2 Descriptors

30.5.2.1 Data Representation in URAM

The DMAC transfers data for transmission and received data to and from the application software via the URAM.

The memory in the URAM for use by the DMAC is configured with control structures referred to as descriptors and associated areas to which the frame data are allocated. Dividing the memory into a control area and data area allows the flexible allocation of frame data to the URAM. This enables sharing of the areas to which frame data are allocated and the use of non-contiguous areas. Frame data can be copied without using the CPU. Arbitration that ensures hardware and software access to the memory area is also available without access to registers of the DMAC.

Figure 30.3 shows an example of the memory maps for descriptors and the descriptor data area in the URAM.

A descriptor consists of its type (DESCR.DT), which controls the descriptor functions, a descriptor pointer (DESCR.DPTR) indicating the start address for storage of the frame data in the descriptor area, and the data size field (DESCR.DS), indicating the amount of frame data. Post-processing interrupt generation can be set up for each descriptor. Enabling and disabling of the interrupt is controlled by the descriptor processed interrupt enable bits (DESCR.DIE).

The descriptor may also hold information related to content. This information does not affect general descriptor functions. It provides information other than the frame data proper, such as on the state of reception.

For details, see **Section 30.5.3.2, Setting Up Reception Descriptors**, and **Section 30.5.4.1, Setting Up Transmission Descriptors**.

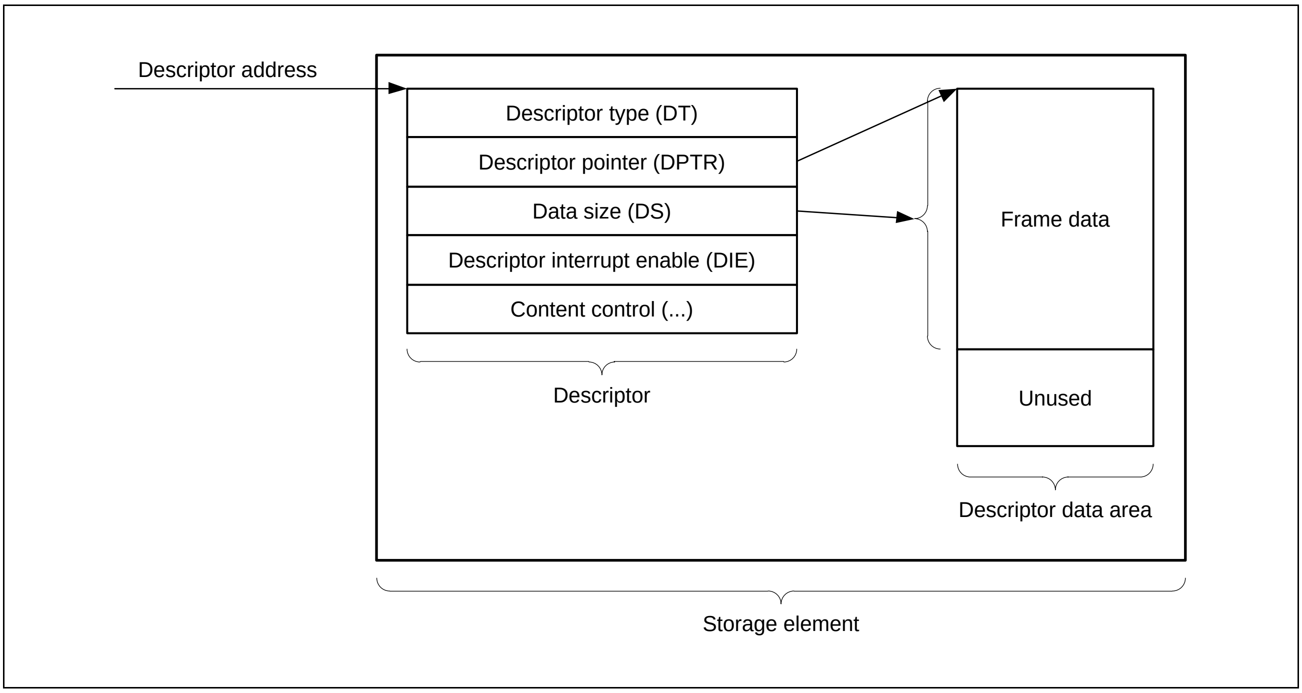


Figure 30.3 Outline of Storage Element Used for Receive and Transmit Queues

The descriptor must be aligned with a 32-bit boundary in the URAM.

Descriptors are configured of 64 bits.

The frame data must also be aligned with a 32-bit boundary in the URAM.

The amount of data in the frame is defined by the data size bits (DESCR.DS). In reception, these bits indicate the upper limit on the size of frames to be received. If the data size is not aligned with a 32-bit boundary, the bytes to the next 32-bit boundary in the data area will be an unused area.

30.5.2.2 Using Descriptor Chains in Queues

Transmission and reception descriptors in the URAM are grouped into queues. The queue is capable of controlling one or more frames. Accordingly, multiple descriptors can be assigned to one queue. A combination of multiple descriptors is referred to as a descriptor chain.

For a descriptor chain, the three general descriptor types listed below are defined. For details on these descriptor types, see **Section 30.5.2.6, Descriptor Type**.

- Descriptors that define frame data
- Descriptors that control the descriptor chain itself (e.g. LINK, EOS).
- Descriptors that arbitrate access by hardware or software

Figure 30.4 shows the two basic topologies for descriptor chains. In the simplified examples in the figure, all descriptors allocated to the chain are stored in the array.

- For a linear descriptor chain, the last descriptor in the array is a control descriptor indicating the end of the descriptors (e.g. EEMPTY).

- For a cyclic descriptor chain, the last descriptor in the array is a control descriptor that returns to the first descriptor in the array (e.g. LINK).

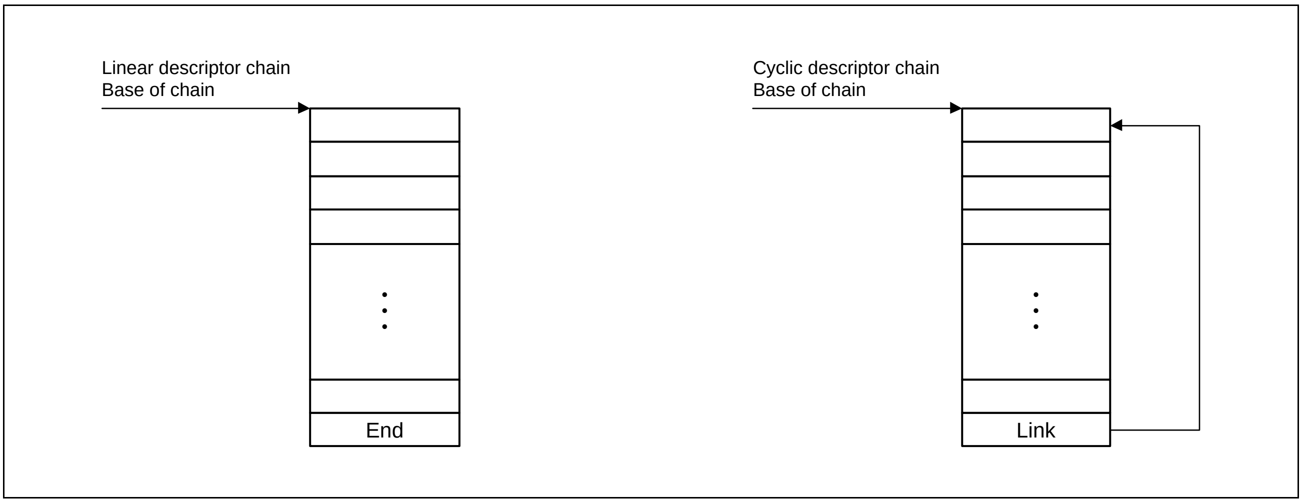


Figure 30.4 Outline of the Basic Descriptor Chains

The relationship between queue and descriptor chain is defined by the base addresses of chain. A queue is connected to one descriptor chain over one round of processing. There is also a method of switching to a different chain while in OPERATION mode.

There are no restrictions on the number of link descriptors and their locations within the chain. The last descriptor of a designed chain determines the topology.

Which chain structure should be used or which topology is suitable depend on the application. A description of how to design descriptor chains to suit various applications is given in **Section 30.5.3.2, Setting Up Reception Descriptors**, and **Section 30.5.4.1, Setting Up Transmission Descriptors**.

30.5.2.3 Descriptor Base Address Table

The base address table in the URAM contains the address of the first descriptor of all chains to be handled by the respective queues.

Entry 0 is used to access transmission queues. Entry 4 is used to access reception queue.

The configuration of entries in the base address table is the same as the configuration of link descriptors. We recommend using the descriptor type (DESCR.DT) LINKFIX. Processing of this link descriptor does not change it, so it does not require updating. The first descriptor of a chain performs hardware and software synchronization. If the application requires hardware and software synchronization for the base addresses, use the descriptor type (DESCR.DT) LINK.

The CPU is only capable of using LINKFIX and LINK as descriptor types (DESCR.DT) of descriptors in the base address table.

Set the location of the base address table in the URAM in the descriptor base address table register (DBAT).

Figure 30.5 shows an example of a base address table for controlling transmission and reception queues. The boxes to the right of the table represent descriptor chains with the desired topologies.

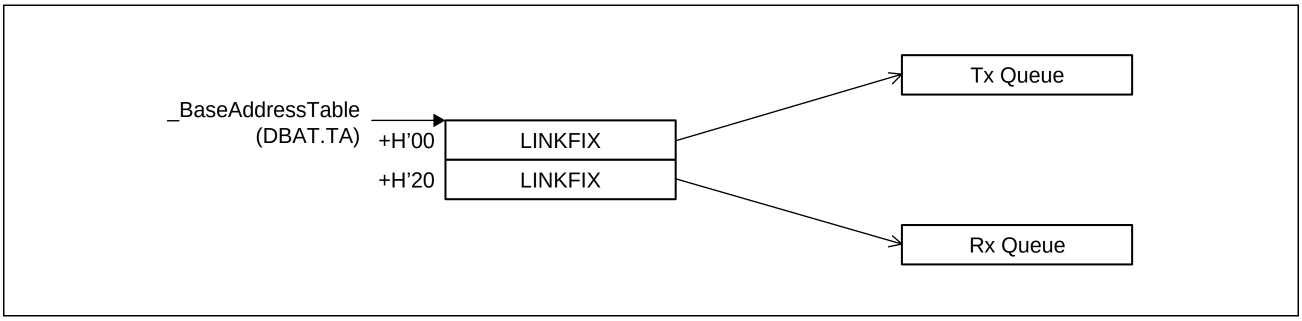


Figure 30.5 Example of a Base Address Table for Reception and Transmission Queues

30.5.2.4 Descriptor Chain Processing

The descriptor that is currently processed or will be processed when the related queue gets active is the current descriptor. The current descriptor address for use by a queue q can be checked in the current descriptor address register q (CDARq).

Current descriptors are stored in registers or in descriptors as described below in the given situations.

- In the descriptor base address table registers for all q queues (DBAT) (DBAT.TA+8*q) when the operating mode shifts to OPERATION mode.
- In the descriptor base address table register (DBAT) (DBAT.TA+8*q) when a base address load request is issued for a queue q by setting the corresponding bit (DLR.LBAq) in the descriptor base address load request register (DLR).
- In DESCR.DPTR for a link descriptor (LINK, LINKFIX) to be processed.
- After a descriptor has been processed, the current descriptor for the same queue is incremented by the size of the descriptors being handled by the queue. The DMAC updates the descriptor type and informs the CPU that the descriptor has been processed.

30.5.2.5 Descriptor Interrupts

A descriptor is able to issue a descriptor interrupt on completion of its processing. The setting of the descriptor interrupt enables bits (DESCR.DIE) in each descriptor selects disabling or generation of the descriptor interrupt.

The descriptor interrupt is a common resource that is shared between reception and transmission queues. Software control of the descriptor interrupt provides a flexible method of application-specific flag processing.

Figure 30.6 illustrates the way in which the DMAC generates descriptor interrupts (or sets bits in the descriptor interrupt status register (DIS.DPFi)). Processing of a descriptor with the value i in the descriptor interrupt enable bits (DESCR.DIE) leads to the corresponding bit in the descriptor interrupt status register (DIS.DPFi) being set.

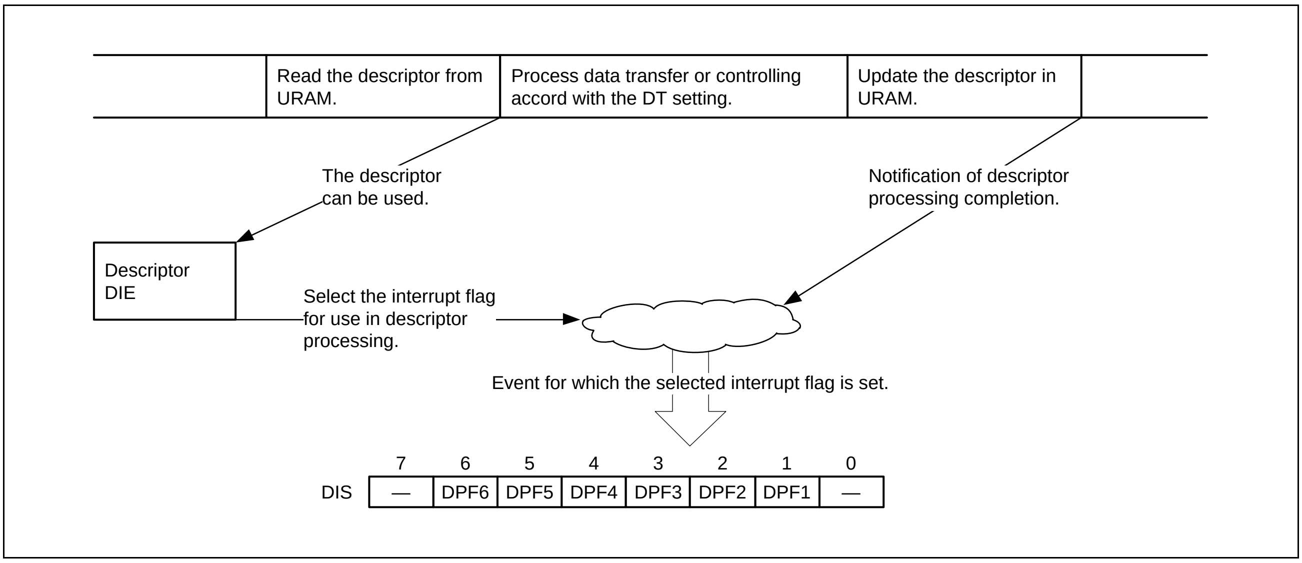


Figure 30.6 Method of Descriptor Interrupt Generation

30.5.2.6 Descriptor Type

The descriptor types (indicated by the DESCR.DT bits) supported by the DMAC fall into the following three categories.

- Definitions of frame data
- Control of descriptor chains
- Hardware and software arbitration

Table 30.7 is a summary of the descriptor types available for the DMAC. Entries under “Name” are the names of the descriptor types and the values under “DT” are the corresponding values to be set in the descriptor type field (DESCR.DT). A given descriptor may be handled differently according to whether it is in a transmission or reception queue, so the transmission and reception columns list the scopes of control and processing of the descriptor types.

The abbreviations defined below are used in the transmission and reception columns.

Definition of SW:

- The descriptor is processed by software.
- Software has access to and may modify the descriptor and descriptor data area.
- This descriptor cannot be changed by hardware (DMAC).

Definition of HW:

- The descriptor is processed by hardware (DMAC).
- Software must modify neither the descriptor nor the descriptor data area.
- Hardware (DMAC) processes this descriptor and subsequently changes the descriptor type.

Invalid:

- This descriptor type is not used in transfer in the given direction (transmission or reception).
- Do not write to this value to the descriptor type (DESCR.DT) field for transfer in the given direction.

- Hardware does not process these descriptor types in the cases listed as invalid. The current descriptor address (CDARq.CDA) will not be changed when processing of a queue for the given direction arrives at a descriptor with this type setting.

Table 30.7 Summary of Descriptor Types

Name	DT	Description	Reception	Transmission
Frame data				
FSTART	5	Frame Start The descriptor points to valid data for a frame. The frame starts with the given data and continues with that indicated by the next descriptor.	SW	HW
FMID	4	Frame Middle The descriptor points to valid data for a frame. The frame started with a previous descriptor and continues to the data indicated by the next descriptor.	SW	HW
FEND	6	Frame End The descriptor points to valid data for a frame. The frame continues from the previous descriptor and ends with the data indicated by in this descriptor.	SW	HW
FSINGLE	7	Frame Single The descriptor points to valid data for a complete frame.	SW	HW
Chain control				
LINK	8	Link Defines the next descriptor in the chain.	HW	HW
LINKFIX	9	Fixed Link Same as LINK, but not changed by DMAC after processing.	SW	SW
EOS	10	End Of Set Control element to split a descriptor chain. The chain stops and waits for user interaction.	HW	HW
HW/SW arbitration				
FEMPTY	12	Frame Empty A descriptor related to frame data but not containing valid data for a frame	HW	SW
DT13	13	Reserved	Invalid	Invalid
DT14	14	Reserved	Invalid	Invalid
FEMPTY_ND	15	Frame Empty No Data storage A descriptor related to frame data but not containing valid data for a frame. The descriptor is processed in the same way as FEMPTY, but data are not stored in the URAM.	HW	Invalid
LEMPTY	2	Link Empty A link descriptor for processing by the DMAC	SW	SW
EEMPTY	3	EOS Empty An EOS descriptor for processing by the DMAC	SW	SW
DT0	0	Reserved	Invalid	Invalid
DT1	1	Reserved	Invalid	Invalid
DT11	11	Reserved	Invalid	Invalid

30.5.2.7 Layout of General Descriptors in the URAM

The DMAC updates processed descriptors in the URAM. The field to be changed in the descriptor being updated depends upon whether the direction is transmission or reception and the queue mode. Other fields will not be changed. There are no restrictions on the values set in unused descriptor fields (indicated by "—" in the figure).

(1) Frame Data Descriptors

The allocation of bits in the frame data descriptors (FSTART, FMID, FEND, and FSINGLE) is shown below.
usable in both reception and transmission

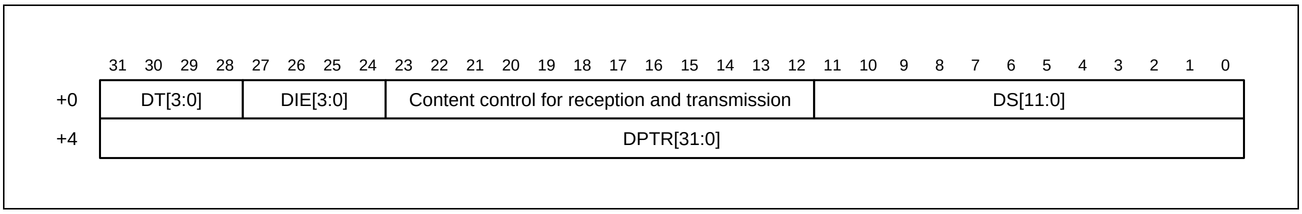


Figure 30.7 The allocation of bits in the frame data descriptors

Table 30.8 Contents of Frame Data Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 5: FSTART 4: FMID 6: FEND 7: FSINGLE For details, see **Section 30.5.3.2, Setting Up Reception Descriptors**, and **Section 30.5.4.1, Setting Up Transmission Descriptors**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).
—	Content Control For details, see **Section 30.5.3.2, Setting Up Reception Descriptors**, and **Section 30.5.4.1, Setting Up Transmission Descriptors**.
DS[11:0]	Data Size Size of the data area/frame data for the descriptor (in bytes)
DPTR[31:0]	Descriptor Pointer Pointer to the data area for the descriptor

Note: Register an address aligned with a 32-bit boundary as the descriptor pointer (DESCR.DPTR).

(2) Hardware/Software Arbitration Descriptors (Only for Reception)

The allocation of bits in the descriptors for hardware/software arbitration (FEMPTY and FEMPTY_ND) is shown below.

The allocation of bits in the arbitration descriptors for use in reception is the same as in frame data descriptors.

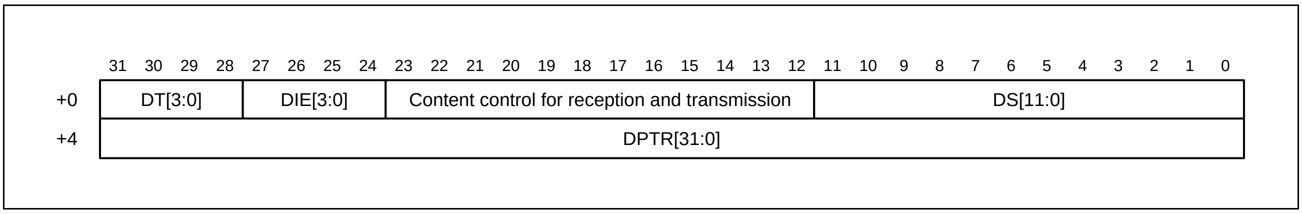


Figure 30.8 The allocation of bits in the arbitration descriptors

Table 30.9 Contents of Hardware/Software Arbitration Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 12: FEMPTY 15: FEMPTY_ND For details, see **Table 30.7, Summary of Descriptor Types**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).
—	Content Control For details, see **Section 30.5.3.2, Setting Up Reception Descriptors**, and **Section 30.5.4.1, Setting Up Transmission Descriptors**.
DS[11:0]	Data Size Size of the data area/frame data for the descriptor (in bytes)
DPTR[31:0]	Descriptor Pointer Pointer to the data area for the descriptor

Note: Register an address aligned with a 32-bit boundary as the descriptor pointer (DESCR.DPTR).

In an FEMPTY descriptor, the descriptor type (DT), descriptor interrupt enable (DIE), data size (DS), and descriptor pointer (DPTR) fields are used.

In an FEMPTY_ND descriptor, the descriptor type (DT), descriptor interrupt enable (DIE), and data size (DS) are used.

(3) Link Descriptors

The allocation of bits in the link descriptors (LINK and LINKFIX) is shown below.

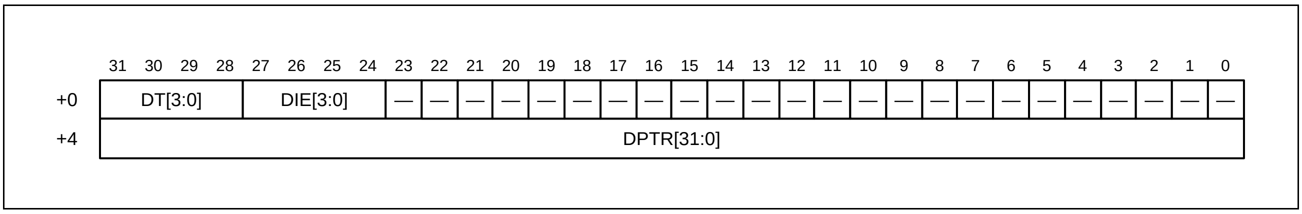


Figure 30.9 The allocation of bits in the link descriptors

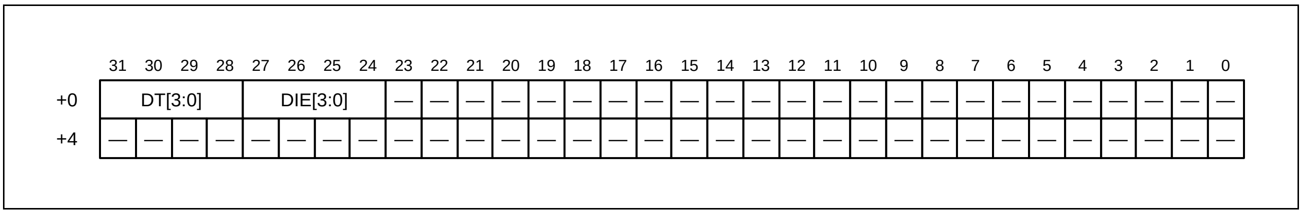
Table 30.10 Contents of Link Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 8: LINK 9: LINKFIX For details, see **Table 30.7, Summary of Descriptor Types**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).
DPTR[31:0]	Descriptor Pointer Pointer to the data area for the descriptor Register an address on a 32-bit boundary.

Note: Register an address aligned with a 32-bit boundary as the descriptor pointer (DESCR.DPTR).

(4) Other Descriptors

The allocation of bits in the other descriptors (EOS, FEMPTY (only for transmission), LEMPTY, and EEMPTY) is shown below.



	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0	DT[3:0]				DIE[3:0]				—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
+4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Figure 30.10 The allocation of bits in the other descriptors

Table 30.11 Contents of Other Descriptors (DESCR)

Bit Name	Function
DT[3:0]	Descriptor Type 10: EOS 12: FEMPTY (only for transmission) 2: LEMPTY 3: EEMPTY For details, see **Table 30.7, Summary of Descriptor Types**.
DIE[3:0]	Descriptor Interrupt Enable 0000b: Descriptor interrupt is disabled. 0001b to 1111b: The corresponding descriptor interrupt is generated (DIS.DPFi).

30.5.2.8 How to Use Frame Data Descriptors

In general, Ethernet frames are not of uniform length. The DMAC is capable of dividing frame data into multiple descriptors in order to minimize the memory capacity for frame data. This function allows processing of frames that are longer than the limit for descriptor data areas. Division can also be applied to frames on the basis of their data structures.

To handle frames and descriptors, four descriptor types (DESCR.DT) as FSTART, FEND, FMID and FSINGLE are defined.

Figure 30.11 shows the mapping of frame data by frame data descriptors. The descriptor data areas are allocated to the URAM. For frames that require division into four or more data areas, additional FMID descriptors can be added as required.

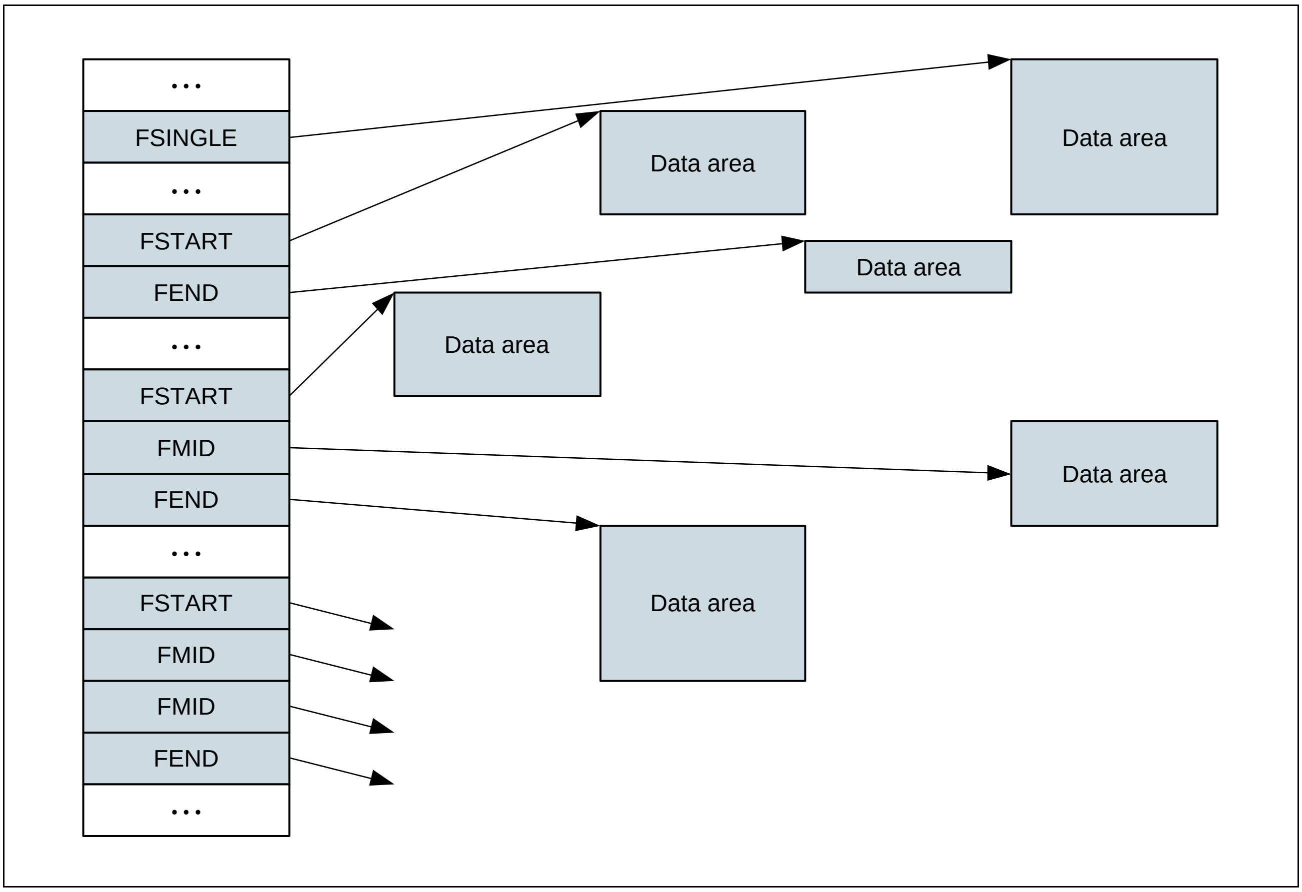


Figure 30.11 Mapping of Frame Data

For reception, the DMAC will store received frame data in the given area. If a received frame has more data than the descriptor data size (DESCR.DS), the DMAC will divide the data up.

For transmission, set the descriptor data size (DESCR.DS) to the actual data size. The DMAC modifies the descriptor type (DESCR.DT) to FEMPTY after processing the relevant descriptor. The data size (DESCR.DS) and descriptor pointer (DESCR.PTR) fields retain their settings.

30.5.2.9 How to Use Chain Control Descriptors

(1) Link Descriptors

The link descriptors can be used to set up cyclic descriptor chains (for details, see **Section 30.5.2.2, Using Descriptor Chains in Queues**)

After a LINK descriptor is processed, its descriptor type (DESCR.DT) is changed to LEMPTY. The descriptor pointer (DESCR.PTR) retains its setting.

After processing of a LINKFIX descriptor, the descriptor type (DESCR.DT) is not updated. Software can change the descriptor type (DESCR.DT), descriptor interrupt enable (DESCR.DIE), and descriptor pointer (DESCR.DPTR). However, DESCR.DT should not be modified by software. Take care to check the current descriptor address register (CDARq.CDA) before changing the descriptor pointer (DESCR.DPTR).

(2) EOS Descriptor

Use the EOS descriptor to divide a descriptor chain into various segments. The queue can continue even after an EOS descriptor.

In transmission, the response to an EOS descriptor is clearing of the transmit start request bit in the transmit configuration control register (TCCR.TSRQ) to 0.

In reception, the response is generation of a receive queue full interrupt (RIS2.QFF), although if the frame currently being received is being divided for storage (received data such as those where some storage is in FMID- or FEND-type frames), the data are not completely stored.

30.5.2.10 How to Use Hardware and Software Arbitration Descriptors

In hardware processing of descriptors, the empty descriptor types (FEMPTY, LEMPTY, and EEMPTY) are used to distinguish various descriptors. For software, they can be used to initiate checking for empty spaces, etc.

(1) FEMPTY and FEMPTY_ND

These descriptor types (DESCR.DT) are used for descriptors that do not contain effective data. Of these, only FEMPTY is used in transmission. The descriptor pointer of a FEMPTYxxx descriptor refers to a descriptor data area.

(2) LEMPTY

This descriptor type (DESCR.DT) is assigned to LINK descriptors after they have been processed. The descriptor pointer (DESCR.DPTR) of an LEMPTY descriptor still points to the linked descriptor.

(3) EEMPTY

This descriptor type (DESCR.DT) is assigned to EOS descriptors after they have been processed. The descriptor pointer (DESCR.DPTR) of an EEMPTY descriptor is not used.

30.5.2.11 Synchronization between Descriptor Access by Hardware and Software

The allocation of descriptor types (DESCR.DT) to the URAM can be used to set up the primary synchronization between hardware and software. By this, the number of CPU accesses to registers of Gigabit Ethernet Interface can be minimized and performance can be increased.

Basic concepts of synchronization:

- Each descriptor type in the set is exclusively for processing by hardware or software, depending on the direction of transfer (see **Table 30.7, Summary of Descriptor Types**).
- Software must not change a descriptor assigned to hardware processing (the hardware does not change descriptors assigned to software processing).

In the case of software processing, the software must process the information in the descriptor and the corresponding frame data before changing the descriptor type. If a descriptor type for hardware is set in DESCR.DT, the software should not change any part of the descriptor or of the corresponding frame data.

30.5.2.12 Tips for Optimizing Performance in Handling Descriptors

The following items are recommended as ways to ensure the optimal use of data structures in the URAM.

They are not requirements but using a different approach may increase the load on the system bus within the LSI chip.

- Register descriptors with 64-bit alignment.
- While in OPERATION mode, use LINKFIX instead of LINK whenever a descriptor need not be changed. Hardware modifies the descriptor type (DESCR.DT) fields of LINK descriptors.
- Frame data is accessed in blocks up to 128 bytes.
- The number of 128 byte borders (addresses H’xxx00 and H’xxx80) and frame data inside should be minimized.
- Design the descriptor chains in ways that minimize parallelism of processing. This helps in dividing the chains into segments allocated to different cache pages, and in arranging the different segments exclusively for access by software or hardware.
- Minimize the number of divided frames. This can reduce the overhead of descriptor handling.

30.5.3 Reception Control

The point of the DMAC is to transfer data between the TOE/E-MAC and URAM without intervention by the CPU.

DMAC needs descriptors that define the amounts of frame data to be stored and the locations. When the E-MAC received a frame, it is transferred to TOE and the conditions of reception is also transferred to DMAC as the E-MAC state. The checksum of received frame is calculated by TOE, the received frame is stored to Reception FIFO temporary. If TOE detects any errors from received frame (such as it was unsupported frame or checksum error), the error information is notified to DMAC. After that, the received frame is transferred to URAM by DMAC. At the time, MAC-Status and Error Information of TOE are written to Reception Descriptor by Write Back. For a description of how to set up descriptors for use in reception, see **Section 30.5.3.2, Setting Up Reception Descriptors**.

Figure 30.12 shows the reception data bus.

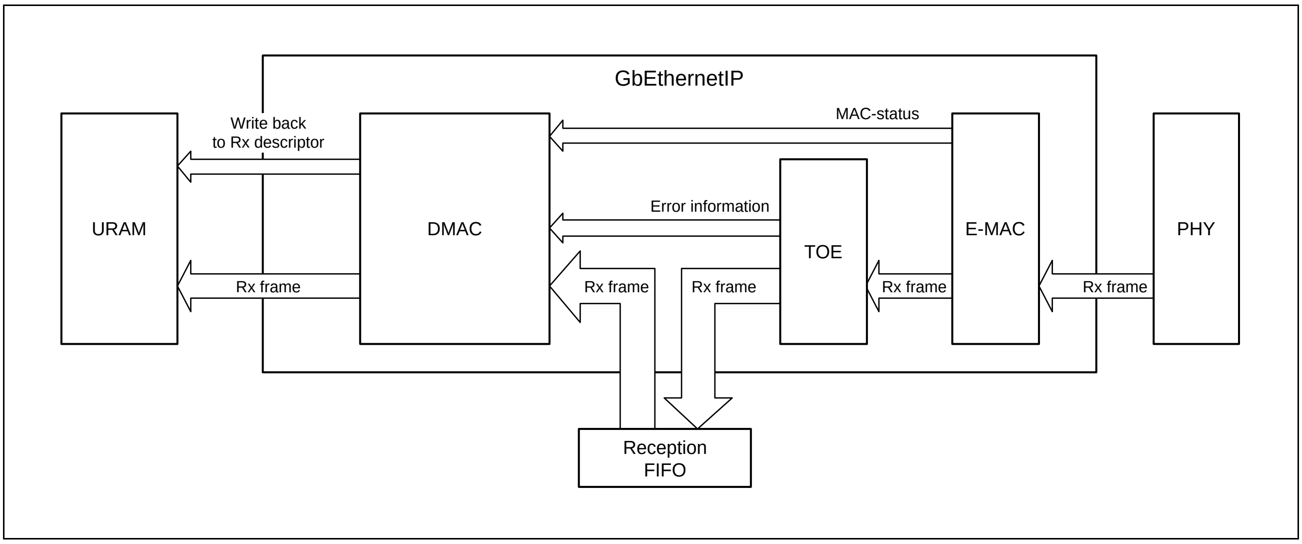


Figure 30.12 Mechanism of General Reception

30.5.3.1 Reception Queues

The DMAC stores all received frames in the URAM.

There is a condition for the DMAC to discard a received frame.

- Detection of an error during reception by the TOE and E-MAC
  - Whether error frames are discarded or stored in reception queue depends on the setting of the error frame enable bit in the receive configuration register (RCR.EFFS). If error frames are to be stored (RCR.EFFS = 1*[1]), they are always stored in queue. In this case, characteristics specific to the queue (e.g. truncation) will vary.

Note 1. RCR.EFFS = 1 is only for debug. Do not set this condition during normal operation.

The flowchart in **Figure 30.13** shows how the DMAC selects the reception queue in accord with the frame type. The result is storage of the frame in the proper queue or the frame being discarded.

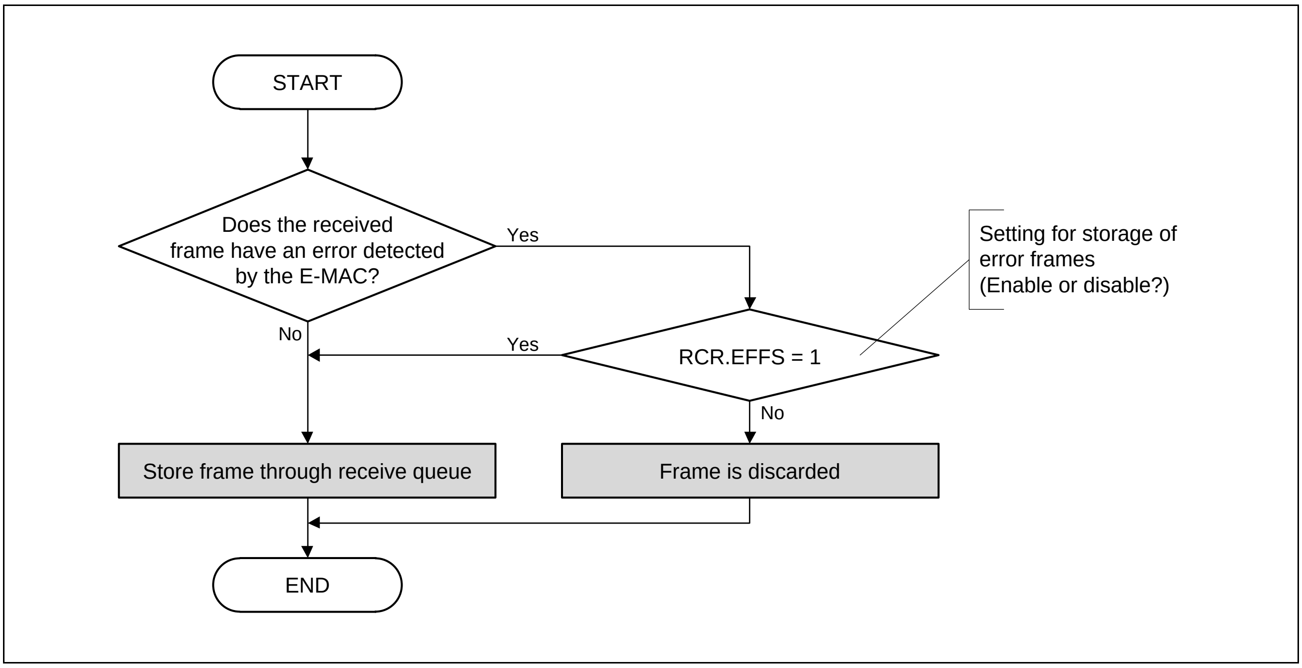


Figure 30.13 Mechanism of Reception Queue Selection

30.5.3.2 Setting Up Reception Descriptors

For reception, the descriptor mechanism is essentially as described in **Section 30.5.2, Descriptors**.

This section describes memory operations that are especially required in handling reception queues.

(1) Reception Descriptor Type

The type of a descriptor is defined by the descriptor type (DESCR.DT) field.

Table 30.12 shows the descriptor types used in reception.

Table 30.12 Descriptor Types in Reception

Descriptor Type (DESCR.DT)	Operation	Write-back
Frame Start (FSTART)	Condition for data not being stored in a reception queue: The RIS2.QFF bit indicates that queue r is full, and the received frame is not stored. Descriptor processing proceeds again in response to further reception.	Not changed
Frame Middle (FMID)	Same as FSTART	Not changed
Frame End (FEND)	Same as FSTART	Not changed
Frame Single (FSINGLE)	Same as FSTART	Not changed
Link (LINK)	Processing proceeds to the descriptor specified by DESCR.DPTR.	LEMPTY
Fixed Link (LINKFIX)	Same as LINK	Not changed
End Of Set (EOS)	A stop point defined by software has been reached. A descriptor of this type within a divided frame (writing of FMID or FEND) stops the frame being stored and the frame is lost. RIS2.QFF indicates that the frame has been lost. If this happens at the start of a frame (writing of FSTART or FSINGLE), storing of frames starts from the next descriptor. In either case, processing shifts to the next descriptor in the chain.	EEMPTY
Frame Empty (FEMPTY)	The descriptor can be used to store received data. Up to DESCR.DS bytes are stored in the descriptor data area. For details, see **Section 30.5.3.3(1), Storing Frame Data in the Descriptor Data Area**.	FSTART, FMID, FEND, or FSINGLE
Frame Empty No Data storage (FEMPTY_ND)	The descriptor can be used to store received data. Up to DESCR.DS bytes are captured from the Reception FIFO but not stored. After processing, DESCR.DS is written back as 0.	FSTART, FMID, FEND or FSINGLE
Link Empty (LEMPTY)	Same as FSTART	Not changed
EOS Empty (EEMPTY)	Same as FSTART	Not changed

(2) Configuration of Reception Frame Data Descriptors

Figure 30.14 shows the configuration of descriptors for use with reception queues. The reception-specific fields (DESCR.MSC, DESCR.PS, DESCR.EI, and DESCR.TI) are described in **Table 30.8**.

For the other fields and the descriptor types, see **Section 30.5.2.6, Descriptor Type**.

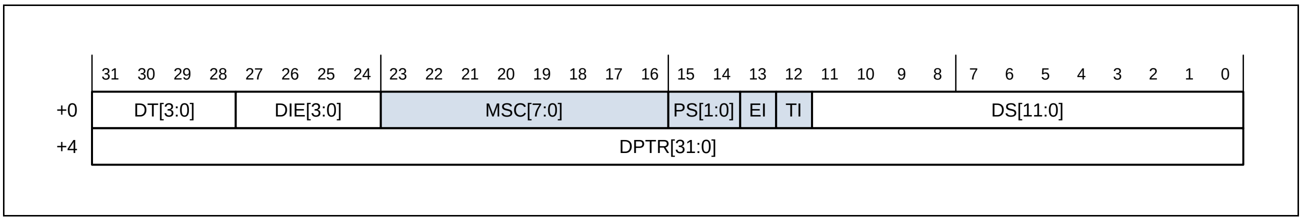


Figure 30.14 Configuration of Descriptor for a Received Frame

Table 30.13 Configuration of a Received Descriptor

Bit Name	Function
MSC	E-MAC/TOE Status Code These bits indicate errors in reception detected by the E-MAC and TOE. In the case of a divided frame, the bits are set to the same value within all descriptors for the frame data. Details of the bits are as follows. MSC [7]: Received frame has a multicast address. (E-MAC) MSC [6]: Check-sum error (TOE) MSC [5]: Received Unsupported frame (TOE) MSC [4]: Received frame has alignment error. (E-MAC) MSC [3]: Received frame is too long. (E-MAC) MSC [2]: Received frame is too short (E-MAC) MSC [1]: Error in frame reception (E-MAC) MSC [0]: Received frame has a CRC error. (E-MAC)
PS	Padding Selection 00b: Padding is not to be inserted.
EI	Error Indication This bit indicates the detection of an error in frame data while a frame was being stored. The bit is set to 1 for a descriptor in which an error has been detected. If the descriptor is for a divided frame, storage of the frame is aborted. 0: No error 1: Error is detected
TI	Truncation Indication This bit indicates whether frame data received from the E-MAC have been truncated before being stored. The bits are set to the same value within all frame data descriptors for a divided frame. 0: Data have not been truncated.

Note: The RCR.EFFS bit specifies whether frames with errors detected by the TOE or E-MAC are to be stored in the URAM or not. When the storing of error frames is disabled, error codes are not written to DESCR.MSC.

30.5.3.3 Reception Processing

After initialization, the DMAC can store received frames in the data area in the URAM as indicated by the descriptor. The DMAC continues to store received data in the URAM as long as space is available for descriptors and data areas.

If the Reception FIFO contains even one frame, storing is executed to the reception queue.

If there is even one empty data descriptor in a queue for which reception has started, the storage of frame data starts. Received frames for a queue that is already full (there is no empty frame descriptor) are discarded from the Reception FIFO.

(1) Storing Frame Data in the Descriptor Data Area

Frame data for storage are assumed to be in either of the two patterns described below.

- The data for an entire frame will fit in the descriptor data area.
  - In this case, the descriptor type (DESCR.DT) is FSINGLE.
- Frame data to be stored in the descriptor data area arrive in divided form.
  - In this case, FSTART is written to the descriptor type (DESCR.DT) bits of the first descriptor of the frame data to arrive and FMID and FEND are written to the type bits of descriptors for subsequent data.

The descriptor type is updated by the DMAC in the last step of descriptor processing, so software can always access the descriptor assigned to DESCR.DT.

The CPU can write FEMPTY (or FEMPTY_ND) directly to the descriptor type field after processing the stored element. Do not change the descriptor or any part of the descriptor data area after FEMPTY (or FEMPTY_ND) is written to DESCR.DT.

(1) Storing Frame Data for a Whole Single Frame

For a frame with an FSINGLE descriptor, all data for the frame are held at the position defined by DESCR.DPTR. DESCR.DS indicates the length of the received frame.

If DESCR.DS is bigger than the actual size of a received frame, the FSINGLE descriptor is stored in place of the FEMPTY or FEMPTY_ND descriptor after processing.

(2) Storing Frame Data as Divided Frames

Divided frames are handled in the same way as a single frame. A frame stored with divided descriptors must be recombined before use. DESCR.EI is only valid in the last descriptor of the sequence for a divided frame.

NOTE

If the data area size setting in DESCR.DS is not a multiple of four, the number of bytes set in DESCR.DS is fetched from the Reception FIFO and the remaining bytes are used as the next storage area.

After a received frame is divided into different descriptors, each storage element is handled separately, and the descriptor type is assigned by software after processing. Accordingly, an error frame (FEMPTYxxx instead of FMID or FEND) may exist while a descriptor chain is being processed. In such a case, the CPU must postpone processing of the error frame to the next trigger point.

(3) No Data are Stored

The application specification may lead to some types of received frames being unimportant (for example, when the application only requires stream data from the Ethernet frames). Storing frames in divided form makes separating out the unnecessary parts of Ethernet frames possible.

If part of a divided frame is not required, use the FEMPTY_ND descriptor for that part so that it is not stored in the URAM. Not storing the data negates the need for bandwidth on the data bus, improving the overall performance.

When an FEMPTY_ND descriptor is processed, DESCR.DS is set to 0. This brings the frame data section of the descriptor into agreement with the FEMPTY type. DESCR.DS = 0 is for the unique identification of the descriptor after writing.

(2) Flow of Reception Descriptor Processing

Constructing a descriptor chain requires software (see **Figure 30.15**).

In the example in the figure, the variable SWdescr (software descriptor pointer) is a structure to identify a descriptor being processed. SWdescr must be initialized after OPERATION mode is entered and a descriptor base address load request (DLR.LBAq) is executed (condition for starting the flow of software operations).

The frame_processing() function processes the stored data. The function can use SWdescr.DT to check whether processing of a frame is completed. How frame data are processed differs with the application, so create functions that handle processing in accord with the specification.

The processing section is common to all modes of reception. The number of frames processed in response to each trigger can be restricted. When multiple frames have to be processed in a batch, waiting for individual trigger boxes must be skipped for these frames.

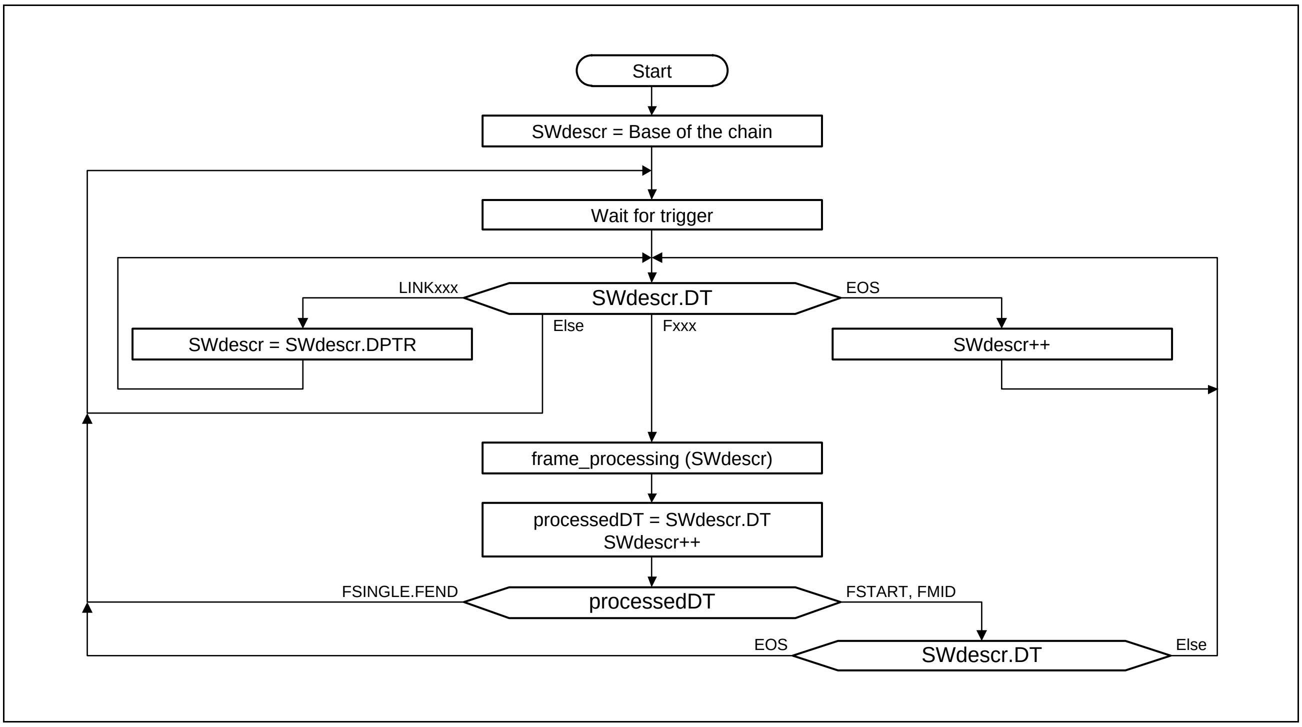


Figure 30.15 Flow of Reception Descriptor Processing

30.5.4 Transmission Control

Areas in the URAM for storing transmission descriptors must also be secured (for descriptors, see **Section 30.5.2, Descriptors**).

The DMAC fetches data from the URAM in accord with the procedure the descriptor describes. The descriptor also retains tag information once the frame has been fetched for transmission. The tag information is used to maintain the relationships between status for the software and the DMAC. The status information for transmitted frames remains accessible after their transmission is completed.

When DMAC started to transfer a frame, it reaches to TOE via Transmission FIFO and TOE FIFO. And after the calculation of transmission frame checksum by TOE, it is transferred to PHY by E-MAC. If any error information is detected by TOE or E-MAC, they are notified to DMAC and stored to MAC Status FIFO/MAC Status FIFO2. The oldest information in them is indicated to MFA/MFA2 of DMAC register.

Figure 30.16 shows the transmission data bus

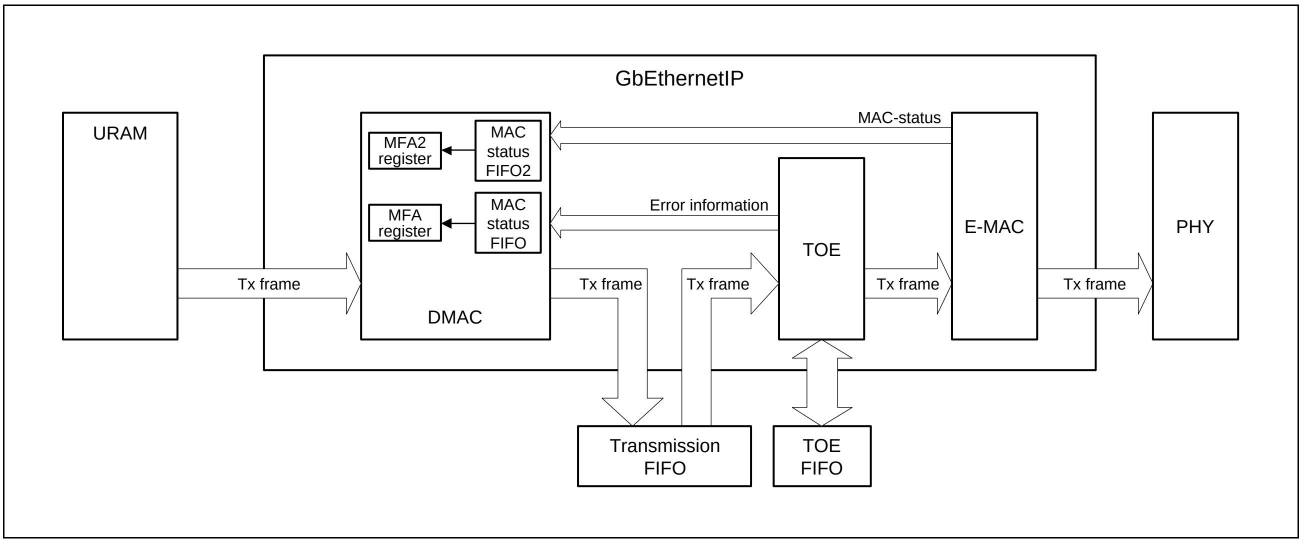


Figure 30.16 Mechanism of General Transmission

30.5.4.1 Setting Up Transmission Descriptors

(1) Transmission Descriptor Type

The type of a descriptor is defined by the descriptor type (DESCR.DT) field.

Table 30.14 shows the descriptor types used in transmission.

Table 30.14 Descriptor Types in Transmission

Descriptor Type (DESCR.DT)	Operation	Write-back
Frame Start (FSTART)	The DMAC fetches the first of the data for the divided frame and proceeds to processing of the next descriptor.	FEMPTY
Frame Middle (FMID)	The DMAC fetches the second or subsequent data for the divided frame and proceeds to processing of the next descriptor.	FEMPTY
Frame End (FEND)	The DMAC fetches the last of the data for the divided frame. The frame of data that has been fetched to the Transmission FIFO is ready for transmission by the E-MAC, and then the DMAC proceeds to processing of the next descriptor.	FEMPTY
Frame Single (FSINGLE)	The DMAC fetches the frame of data. The frame of data that has been fetched to the Transmission FIFO is ready for transmission by the E-MAC, and then the DMAC proceeds to processing of the next descriptor.	FEMPTY
Link (LINK)	Processing proceeds to the descriptor specified by DESCR.DPTR.	LEMPTY
Fixed Link (LINKFIX)	Same as LINK	Not changed
End Of Set (EOS)	This is a transmission stop point defined by software This leads to clearing of the transmit start request bit (TCCR.TSRQ), which stops transmission. When the TCCR.TSRQ is again set to 1 (a new transmission start request is issued), processing proceeds to the next descriptor.	EEMPTY
Frame Empty (FEMPTY)	No frame data are ready for transmission This leads to clearing of the transmit start request bit (TCCR.TSRQ), which stops transmission. When the TCCR.TSRQ is again set to 1 (a new transmission start request is issued), processing starts at this descriptor.	Not changed
Link Empty (LEMPTY)	Same as FEMPTY	Not changed
EOS Empty (EEMPTY)	Same as FEMPTY	Not changed

(2) Configuration of Transmission Frame Data Descriptors

Figure 30.17 shows the configuration of descriptors for use with transmission queues. The transmission-specific fields (DESCR.TSR, and DESCR.TAG) are described in **Table 30.15**.

For the other fields and the descriptor types, see **Section 30.5.2.6, Descriptor Type**.

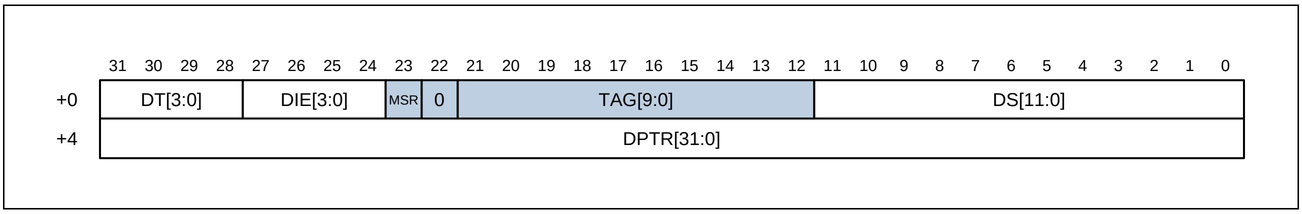


Figure 30.17 Configuration of Descriptor for a Transmitted Frame

Table 30.15 Configuration of a Transmission Descriptor

Bit Name	Function
MSR	MAC Status storage Request This bit configures if MAC transmission status of the frame is stored in MAC status FIFO. 0b: No status information is stored 1b: Status information is to be kept for this frame in MAC status FIFO This bit is considered only if DESCR.DT = FEND or FSINGLE. *Note*: If there is error during transmission of frame then status is always stored.
TAG	Frame Tag This TAG field is used to associate each frame data. Frame TAG is not required but is recommended. This bit is available while the current DESCR.DT is FEND or FSINGLE.

30.5.4.2 Transmission

(1) Transmitting Frames

Setting the transmit start request bit in the transmit configuration control register (TCCR.TSRQ) starts the transfer of frames from the corresponding transmission queue.

The descriptor in the current descriptor address (CDARq.CDA) for the queue (with q = 0) is read first.

If this descriptor is a descriptor for frame transmission (FSINGLE, etc.), the DMAC fetches the frame data from the data area indicated by the descriptor, writes FEMPTY back to the descriptor type (DESCR.DT) bits to indicate completion of this processing, then proceeds to processing of the next descriptor.

If the descriptor is not for transmission, processing is as dictated by the given descriptor (for these descriptors, see the descriptions in **Section 30.5.2, Descriptors**).

If a base address load request is issued for a descriptor chain while it is being processed (by setting 1 in the LBAq bit for transmission queue q that is currently being processed in the descriptor base address load request register, DLR), processing proceeds to the new descriptor chain. Changing the chain does not interrupt frame fetching, but note that frames that have not been fetched from the old chain remain where they are.

Figure 30.18 shows descriptor processing during transmission.

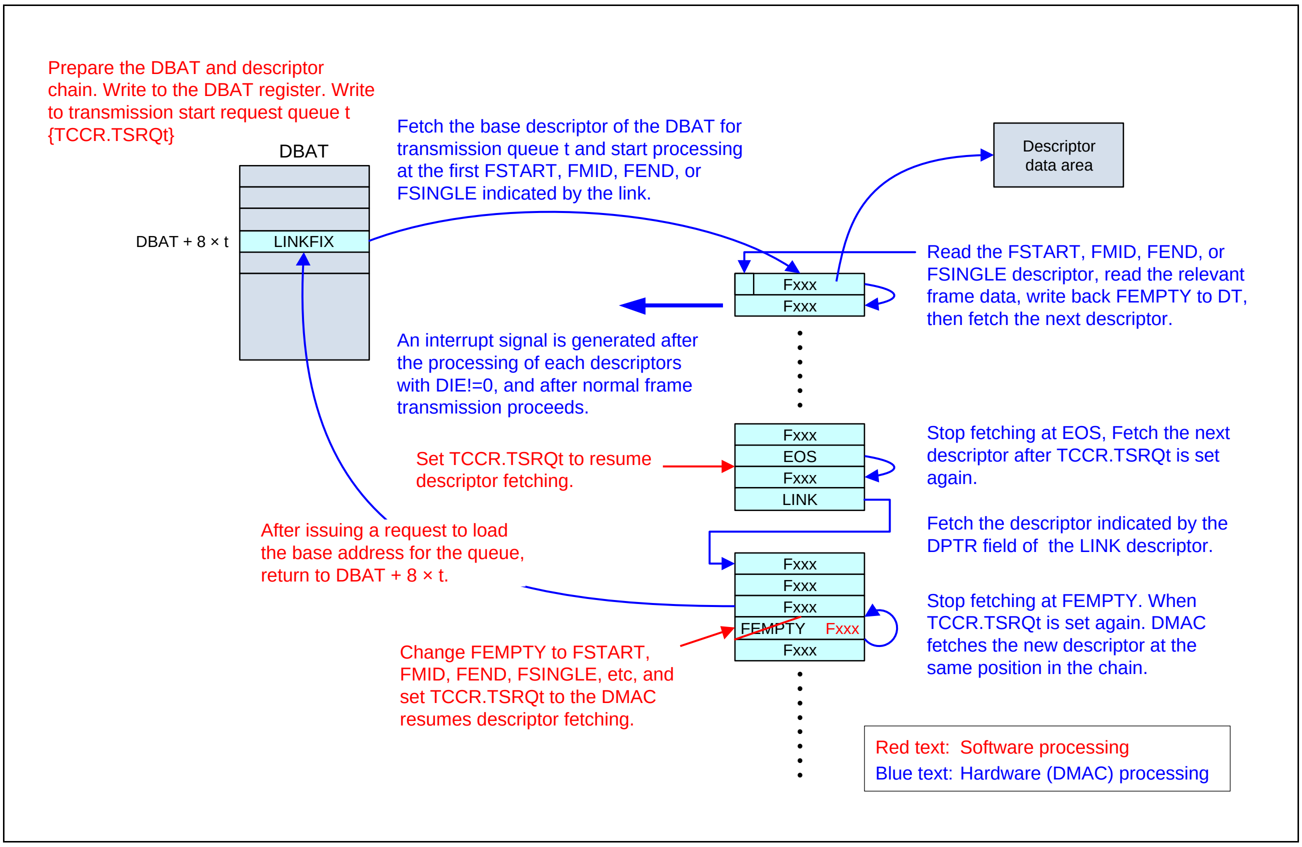


Figure 30.18 Descriptor Processing During Transmission

(2) Examples of Descriptor Usage

(1) Immediate Frame Transmission

Immediate frame transmission is a pattern in which fetching by the DMAC starts whenever software adds data to a queue. FEMPTY descriptors are used as stop points to keep the hardware and software in synchronization.

Create descriptor chains that have FEMTPY descriptors at the stop points.

Figure 30.19 shows the flow for software implementing this pattern. SW should read back written descriptor between changing FEMPTY descriptor before writing to TCCR.TSRQ.

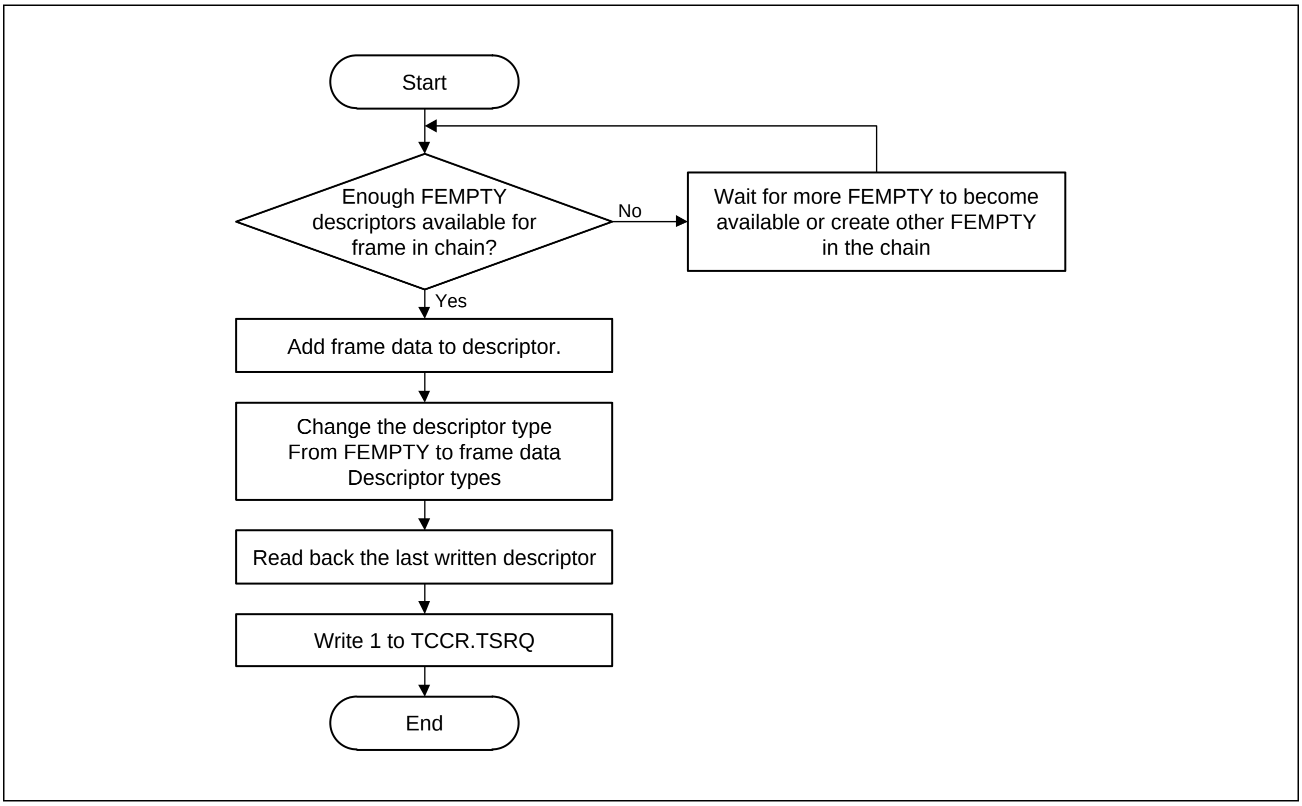


Figure 30.19 Software Flow for Immediate Frame Transmission

Figure 30.20 shows software and DMAC operations for immediate frame transmission.

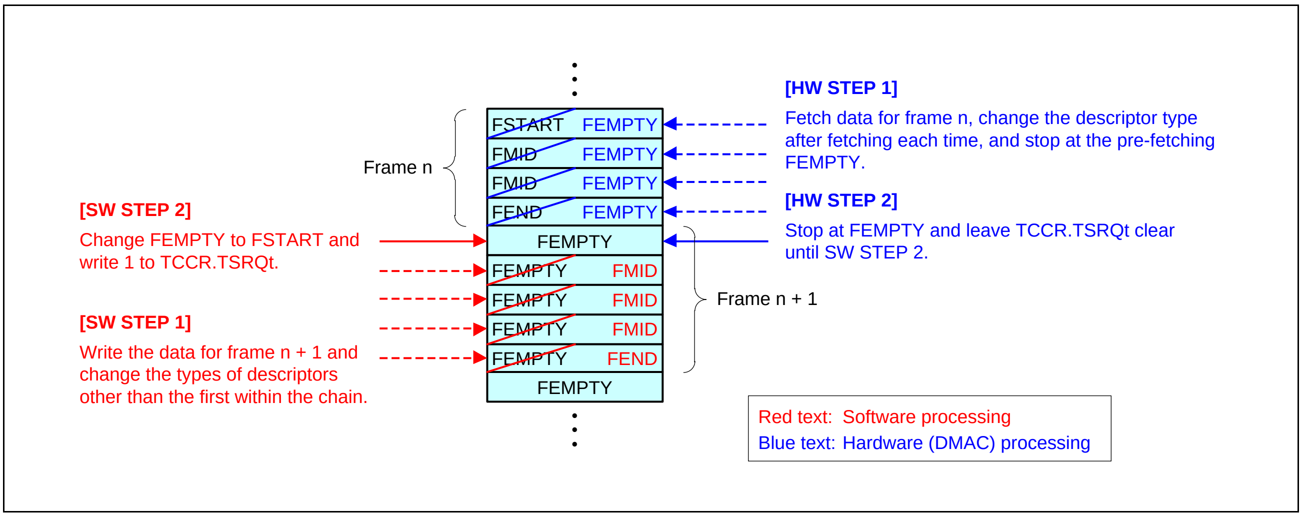


Figure 30.20 Software and DMAC Operations for Immediate Frame Transmission

(2) Frame Set Transmission with Changing of the Active Descriptor Chain

This pattern is used when data are transmitted with a delay for software control to secure bandwidth or for other reasons, rather than immediately transmitted. EOS descriptors are used for the stop points. Start by creating a descriptor chain that has a FEMPTY descriptor as its stop point.

Figure 30.21 shows the software flow in this pattern.

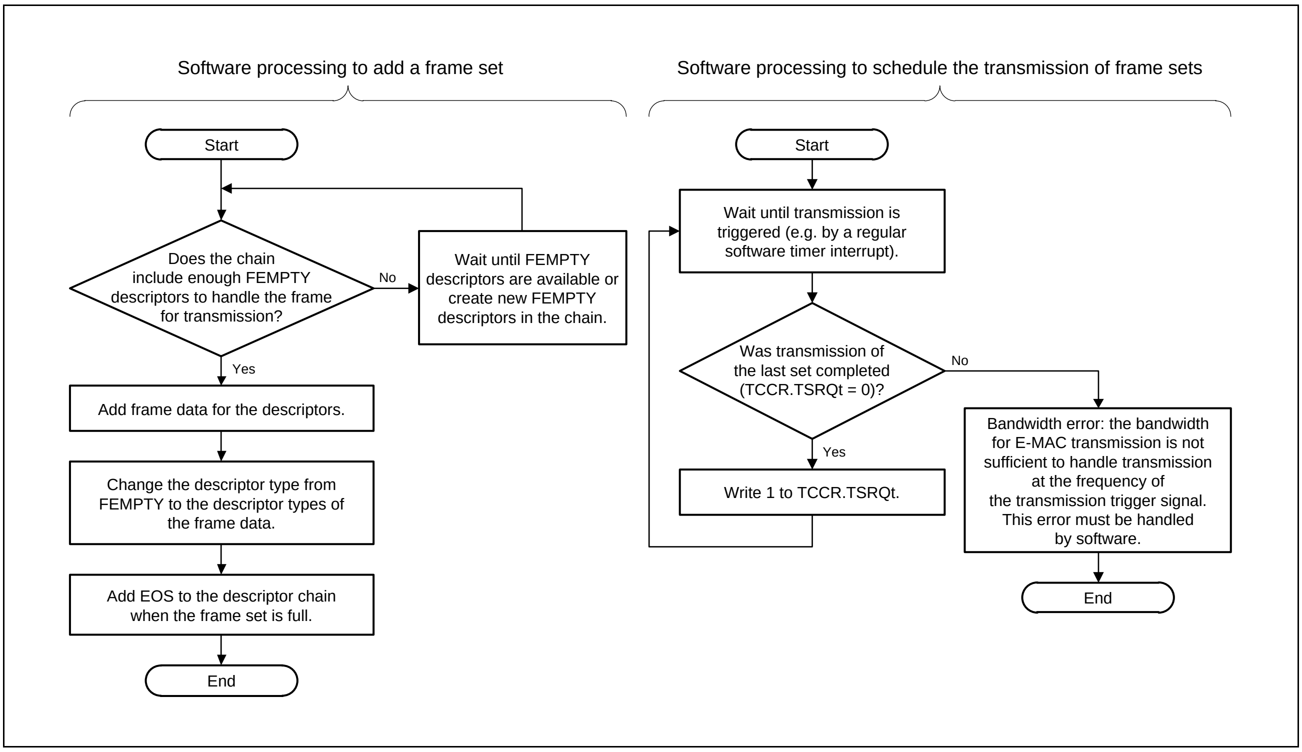


Figure 30.21 Software Flow for Frame Set Transmission with Changing of the Active Descriptor Chain

Figure 30.22 shows software and DMAC operations for frame set transmission.

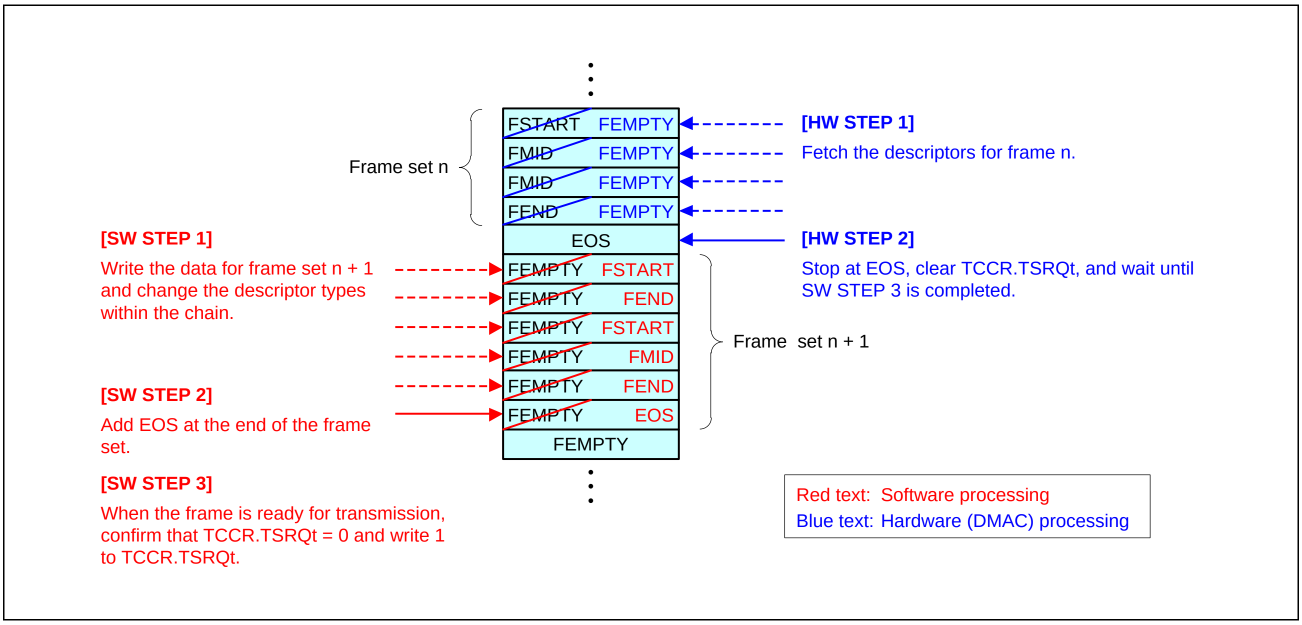


Figure 30.22 SW and DMAC Operations for Frame Set Transmission with Changing of the Active Descriptor Chain

(3) Frame Set Transmission Using a Shadow Descriptor Chain

This pattern is used when data are transmitted with a delay for software control to secure bandwidth or for other reasons, rather than immediately transmitted. Two or more descriptor chains are used. The chains are classified as active or shadow chains. EOS descriptors are used for the stop points.

Create descriptor chains that have FEMPTY descriptors at the stop points.

Figure 30.23 shows the flow for software implementing this pattern.

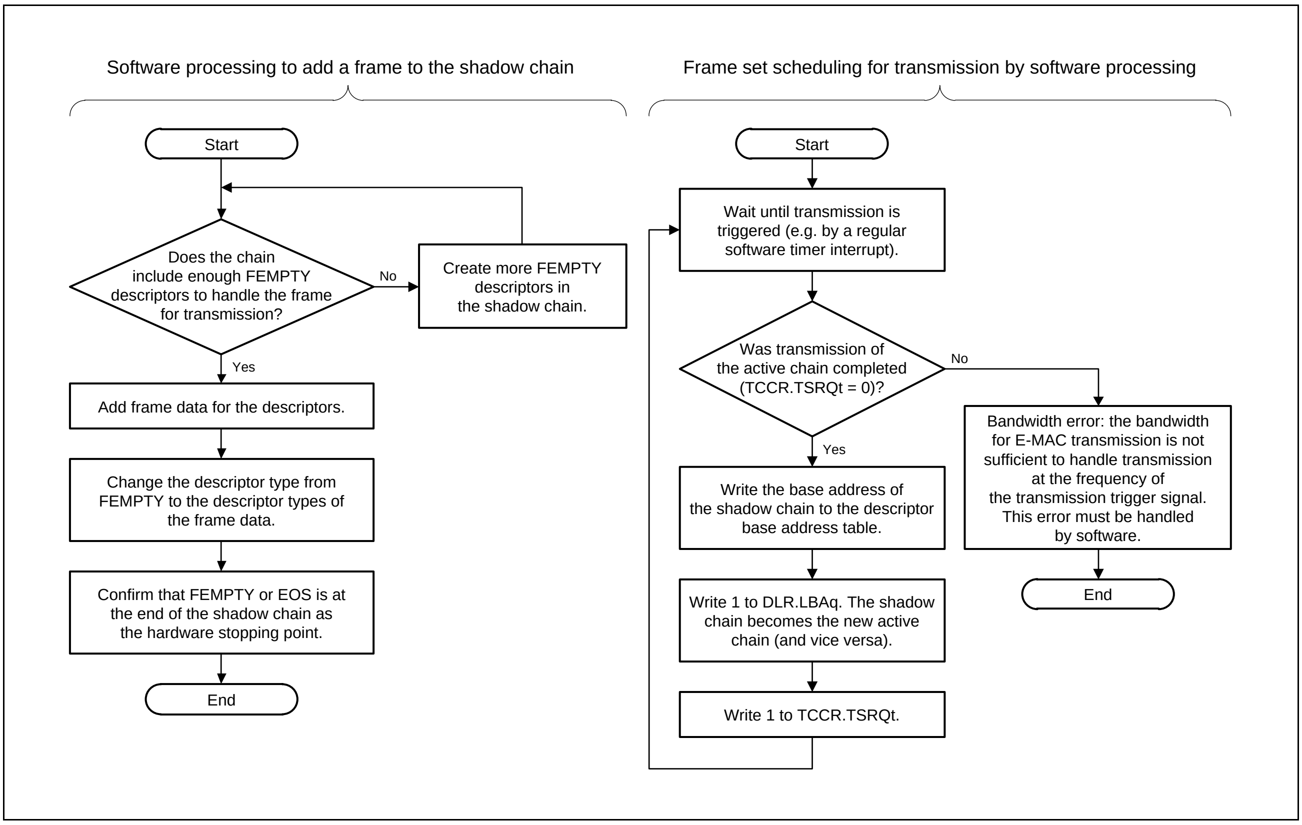
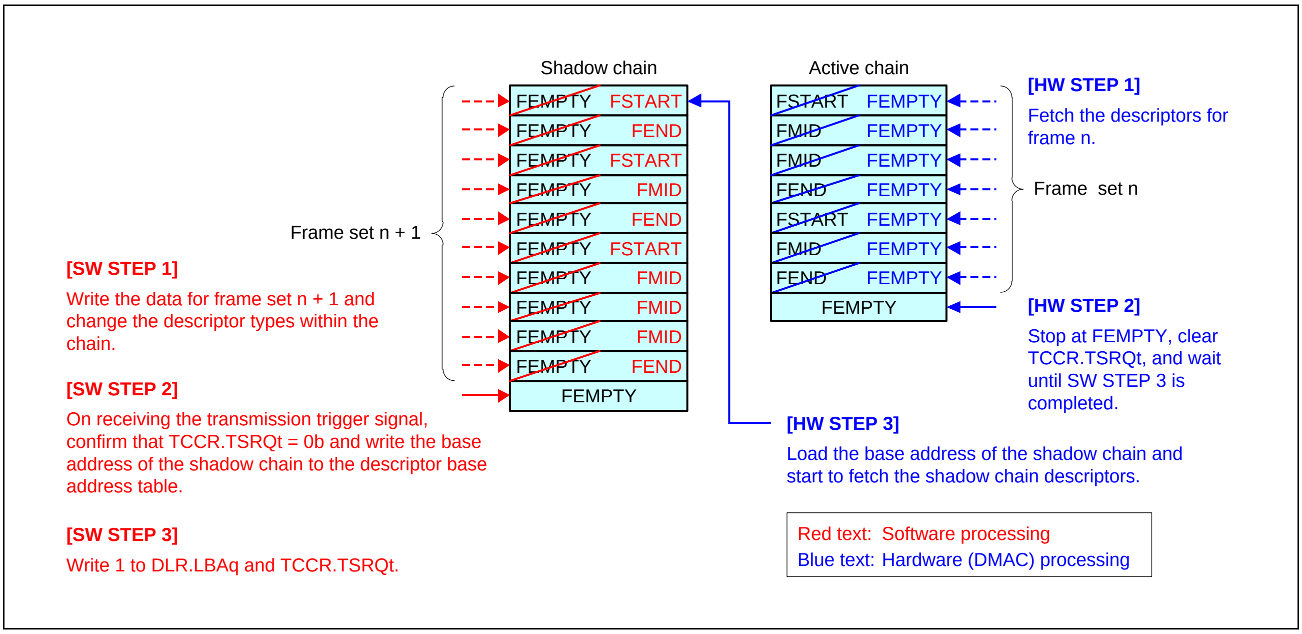


Figure 30.23 Software Flow for Frame Set Transmission Using the Shadow Descriptor Chain

Figure 30.24 shows software and DMAC operations for frame set transmission.



[HW STEP 1]
Fetch the descriptors for frame n.

[SW STEP 1]
Write the data for frame set n + 1 and change the descriptor types within the chain.

[HW STEP 2]
Stop at FEMPTY, clear TCCR.TSRQt, and wait until SW STEP 3 is completed.

[SW STEP 2]
On receiving the transmission trigger signal, confirm that TCCR.TSRQt = 0b and write the base address of the shadow chain to the descriptor base address table.

[HW STEP 3]
Load the base address of the shadow chain and start to fetch the shadow chain descriptors.

[SW STEP 3]
Write 1 to DLR.LBAq and TCCR.TSRQt.

Red text: Software processing
Blue text: Hardware (DMAC) processing

Figure 30.24 SW and DMAC Operations for Frame Set Transmission Using the Shadow Descriptor Chain

30.5.5 Integrity Checking

The DMAC can detect and identify errors produced in the processing of Ethernet frames and in the transfer of frame data for transmission and reception.

30.5.5.1 Integrity Checking in Reception

The aim of integrity checking in reception is preventing the storage of error frames in the URAM. If an error frame is stored in the URAM, software can get information to identify the frame as an error frame.

30.5.5.2 Integrity Checking in Transmission

The purpose of integrity checking in transmission is to prevent the transmission of broken frames.

Since transmission of a frame by the E-MAC can neither be stopped nor disabled once it has started, this check involves intensive monitoring for problems that can arise during fetching.

30.5.5.3 Monitoring in Both Reception and Transmission

(1) Errors in access to the URAM for reading of descriptors

The same descriptor may be processed again because the current descriptor address (CDARq.CDA) was not changed.

If this problem occurs in a divided frame, the sequence may be broken.

In reception

- The received frame will be lost.
- The same problem will occur for the next frame of data received for the same queue.

In transmission

- The transmit start request bit in the transmit configuration control register (TCCR.TSRQ) is set to 0.
- The frame will be lost from the Transmission FIFO.

Errors in access to read descriptors from the URAM are detected from the response signal of the AXI-Bus.

(2) Illegal configuration of a descriptor by an application

The same descriptor may be processed again because the current descriptor address (CDARq.CDA) was not changed.

If this problem occurs in a divided frame, the sequence may be broken.

In reception

- The received frame will be lost.
- The same problem will occur for the next frame of data received for the same queue.

In transmission

- The transmit start request bit in the transmit configuration control register (TCCR.TSRQ) is set to 0.
- The frame will be lost from the Transmission FIFO.

(3) Errors in access to the URAM for writing of descriptors

As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start request bit in the transmit configuration control register (TCCR.TSRQ) are updated.

As DESCR.DT was not updated, hardware and software synchronization may have been destroyed.

Errors in access to write descriptors to the URAM are detected from the response signal of the AXI-Bus.

30.5.5.4 Monitoring in Reception

(1) Errors in access to the URAM for writing of data

- As in the case where no error occurs, the current descriptor address (CDARq.CDA) is updated.
- DESCR.EI is set to indicate incorrect contents.
- This problem occurring in a divided frame may break the descriptor sequence, making the queue unusable.

Errors in access to write data or descriptors to the URAM are detected from the response signal of the AXI-Bus.

(2) Error of the Reception FIFO

- Received frames are all invalidated.
- All frames stored as received frames are discarded. At this time, the number of frames and queue information cannot be captured.

If the Reception FIFO RAM faults, DMAC detects the fault as error of the Reception FIFO.

30.5.5.5 Monitoring in Transmission

(1) Errors in Access for Reading Data from the URAM

- Data that have already been fetched are discarded from the Transmission FIFO.
- When an error of this type occurs during processing of an FSINGLE or FEND descriptor:z
  As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start bit in the transmit configuration control register (TCCR.TSRQ) are updated. Fetching resumes after the error frame.
- When an error of this type occurs during processing of an FSTART or FMID descriptor:
  The current descriptor address (CDARq.CDA) is not updated.
  The transmit start bit in the transmit configuration control register (TCCR.TSRQ) is set to 0.

Errors in access to read data from the URAM are detected from the response signal of the AXI-Bus.

(2) Overflow of the Transmission FIFO

- As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start bit in the transmit configuration control register (TCCR.TSRQ) are updated. Fetching resumes after the error frame.
- The frame will be discarded from the FIFO.

(3) Damaged Data in the Transmission FIFO

- Fetching is not affected by damaged data.
- Since damaged data from the FIFO is only detected during frame transmission, an error frame may be transmitted.
- The information of MAC Status FIFO and MAC Status FIFO2 will be inconsistent.

If damaged data in the Transmission FIFO is an error due to the Transmission FIFO, this is detected by the DMAC.

30.5.6 Checksum Calculation

TOE calculates Checksum of received frames from E-MAC. And then it is outputted to DMAC. TOE also calculates Checksum of transmission frames from DMAC. And then it is outputted to E-MAC.

To use this function, configure E-MAC as CRC Pass Through Mode (CXR20.RCPT = 1).

TOE supports Checksum Calculation for following part of frames. See **Section 30.5.6.4** for about the details of Supported Frames.

- IPv4 Header
- IPv4 TCP/UDP/ICMP
- IPv6 TCP/UDP/ICMP

30.5.6.1 Checksum Calculation handling

(1) Reception Handling

The result of Checksum Calculation is attached to last 4 bytes of Ethernet Frames like **Figure 30.25**. And then the handled frames are transferred to memory by DMAC. If the frame does not have checksum error at the part of IPv4 Header or TCP/UDP/ICMP, the value of “H’0000” is attached to each part as the result of Checksum Calculation. The case of Unsupported Frame, the value of “H’FFFF” is attached. For example, if the part of IP Header is unsupported, “H’FFFF” is set to both field of IPv4 Header and TCP/UDP/ICMP. The case of IPv6, IPv4 Header field is always set to “H’FFFF”. And the set value of TCP/UDP/ICMP field depends on configuration for support conditions.

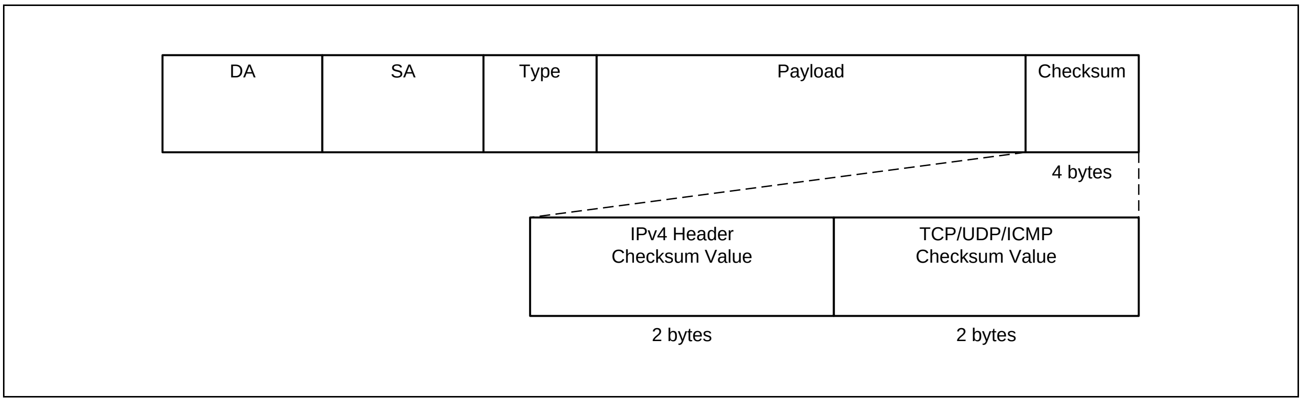


Figure 30.25 The field of checksum attaching field (Ex. Ethernet Frame of DIX)

(2) Transmission Handling

The result of Checksum Calculation is set to the Checksum field of each IPv4 Header/TCP/UDP/ICPM. For the Unsupported Frames, those fields are not changed. If a transmission frame is an UDP frame of IPv4 and its Checksum value in the UDP Header field is “H’0000”, TOE does not calculate Checksum for UDP part of this frame. Because Checksum for IPv4 UDP frame is defined as optional function in standards. If a transmission IPv4 UDP frame does not need to Checksum Value, set “H’0000” to the Checksum field in the UDP Header.

30.5.6.2 Error Notification

When TOE detects any errors by Checksum Calculation, the error information is notified to DMAC. The errors are detected by TOE are followings:

- Unsupported Transmission Frame Error (This error information is indicated to “MFA” register)
- Unsupported Reception Frame Error (This error information is stored to Descriptor by Write Back)
- Reception Frame has Checksum Error (This error information is stored to Descriptor by Write Back)

30.5.6.3 Individually Configuration by the frame type

The Enable/Disable setting of Checksum Calculation is configurable by CSR1 and CSR2 register. And CSR1 and CSR2 register are able to configure Enable/Disable setting individually for frame type. For example, it is possible to disable the checksum calculation for only UDP part of IPv4 frame.

The relationship between the Frame Type and Error Notification to DMAC are indicated at **Table 30.16** and **Table 30.17**. **Table 30.16** is for Transmission Frames. **Table 30.17** is for Reception Frames. The condition of both tables is that the checksum calculation for all frame type is enabled by CSR1 and CSR2 register.

If the checksum calculation of each Frame Type is configured as Disable individually by CSR1 and CSR2 register, the operation is followings. (Following descriptions are for the normal frames)

(1) Disabled Transmission Checksum Calculation

TOE notifies an Unsupported Transmission Frame Error to DMAC. And the Checksum Values which Type is configured as “Disable” are not changed.

(2) Disabled Reception Checksum Calculation

The frames which have Disabled configuration Types are handled as Unsupported Frame. So, the error notification to DAMC is same as the case of Unsupported Frames.

If the other checksum field which is configured as Enable has Checksum Error, Received Checksum Error is notified to DMAC. Received Unsupported Error also is notified. The value of “H’FFFF” is set to the checksum field which is configured as Disable.

Table 30.16 Relationship between Transmission Frame Type and Error Notification to DMAC

Transmission Frame Type						Checksum Operation		
VLAN-tag		IP		Transport protocol		Error Notification to DMAC	Checksum Value	
Number of tags	TPID	Protocol	Condition	Protocol	Condition	Unsupported Frame Error	IPv4 Header	TCP/UDP/ICMP
0 or 1	Nothing or H’8100	IPv4	Not Fragmented Packet	TCP/ICMP	—	Non-Asserted	Calculated Value	Calculated Value
				UDP	Checksum Field is not H’0000	Non-Asserted	Calculated Value	Calculated Value
					Checksum Field is H’0000	Non-Asserted	Calculated Value	Through (H’0000)
				Except TCP/ICMP/UDP	—	Asserted	Calculated Value	—
			Fragmented Packet	don’t care	don’t care	Asserted	Calculated Value	—
		IPv6	Non-Extended Header or The number of Supported Extended Header is from 1 to 8*1	TCP/ICMP	—	Non-Asserted	—	Calculated Value
				UDP	—	Non-Asserted	—	Calculated Value
				Except TCP/ICMP/UDP	—	Asserted	—	—
			The number of Supported Extended Header is 9 or more*1	don’ t care	don’ t care	Asserted	—	—
			Has Unsupported Extended Header*1	don’ t care	don’ t care	Asserted	—	—
		Except IPv4/IPv6	—	don’t care	don’t care	Asserted	—	—
1	Except H’8100	don’t care	don’t care	don’t care	don’t care	Asserted	—	—
2 or more	don’t care	don’t care	don’t care	don’t care	don’t care	Asserted	—	—
Illegal Frames*2						Asserted	—	—

Note 1. See the **Section 30.5.6.4** for about the Supported Extended Headers.

Note 2. The Frame Length is shorter than the result of Frame Analysis by TOE.

Table 30.17 Relationship between Reception Frame Type and Error Notification to DMAC

Reception Frame Type						Field of Checksum Error	Checksum Operation			
VLAN-tag		IP		Transport protocol			Error Notification to DMAC		Checksum Value	
Number of tags	TPID	Protocol	Condition	Protocol	Condition		Checksum Error	Unsupported Frame Error	IPv4 Header	TCP/ UDP/ ICMP
0 or 1	Nothing or H’8100	IPv4	Not Fragmented Packet	TCP/ICMP	—	Non-Error	Non-Asserted	Non-Asserted	H’0000	H’0000
					—	IPv4 Header	Asserted	Non-Asserted	Calculated Value	H’0000
					—	TCP/ICMP	Asserted	Non-Asserted	H’0000	Calculated Value
					—	IPv4 Header and TCP/ICMP	Asserted	Non-Asserted	Calculated Value	Calculated Value
				UDP	Checksum Field is not H’0000	Non-Error	Non-Asserted	Non-Asserted	H’0000	H’0000
						IPv4 Header	Asserted	Non-Asserted	Calculated Value	H’0000
						UDP	Asserted	Non-Asserted	H’0000	Calculated Value
						IPv4 Header and UDP	Asserted	Non-Asserted	Calculated Value	Calculated Value
					Checksum Field is H’0000	Non-Error	Non-Asserted	Non-Asserted	H’0000	H’FFFF
						IPv4 Header	Asserted	Non-Asserted	Calculated Value	H’FFFF
				Except TCP/ ICMP/ UDP	—	Non-Error	Non-Asserted	Asserted	H’0000	H’FFFF
					—	IPv4 Header	Asserted	Asserted	Calculated Value	H’FFFF
			Fragmented Packet	don’t care	don’t care	Non-Error	Non-Asserted	Asserted	H’0000	H’FFFF
				don’t care	don’t care	IPv4 Header	Asserted	Asserted	Calculated Value	H’FFFF
		IPv6	Non-Extended Header or The number of Supported Extended Header is from 1 to 8*1	TCP/ICMP	—	Non-Error	Non-Asserted	Non-Asserted	H’FFFF	H’0000
					—	TCP/ICMP	Asserted	Non-Asserted	H’FFFF	Calculated Value
				UDP	Checksum Field is not H’0000	Non-Error	Non-Asserted	Non-Asserted	H’FFFF	H’0000
						UDP	Asserted	Non-Asserted	H’FFFF	Calculated Value
					Checksum Field is H’0000	—	Asserted	Non-Asserted	H’FFFF	H’FFFF
				Except TCP/ ICMP/ UDP	—	—	Non-Asserted	Asserted	H’FFFF	H’FFFF
			The number of Supported Extended Header is 9 or more*1	don’t care	don’t care	—	Non-Asserted	Asserted	H’FFFF	H’FFFF
			Has Unsupported Extended Header*1	don’t care	don’t care	—	Non-Asserted	Asserted	H’FFFF	H’FFFF
		Except IPv4/IPv6	—	don’t care	don’t care	—	Non-Asserted	Asserted	H’FFFF	H’FFFF
1	Except H’8100	don’t care	don’t care	don’t care	don’t care	—	Non-Asserted	Asserted	H’FFFF	H’FFFF
2 or more	don’t care	don’t care	don’t care	don’t care	don’t care	—	Non-Asserted	Asserted	H’FFFF	H’FFFF
Illegal Frames*2						—	Non-Asserted	Asserted	H’FFFF	H’FFFF

Note 1. See the **Section 30.5.6.4** for about the Supported Extended Headers.

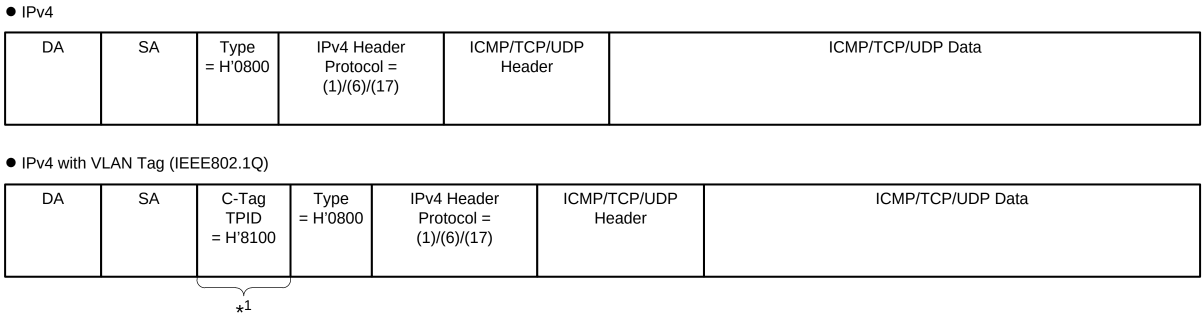
Note 2. The Frame Length is shorter than the result of Frame Analysis by TOE. Or the case of E-MAC detected Errors.

30.5.6.4 Supported Frame for Checksum Calculation

Following Frames are example of Supported/Unsupported Frame Formats for Checksum Calculation.

(1) For IPv4 Checksum Calculation

Figure 30.26 is Supported Frame Format for the Checksum Calculation of IPv4 Header and IPv4 TCP/UDP/ICMP.

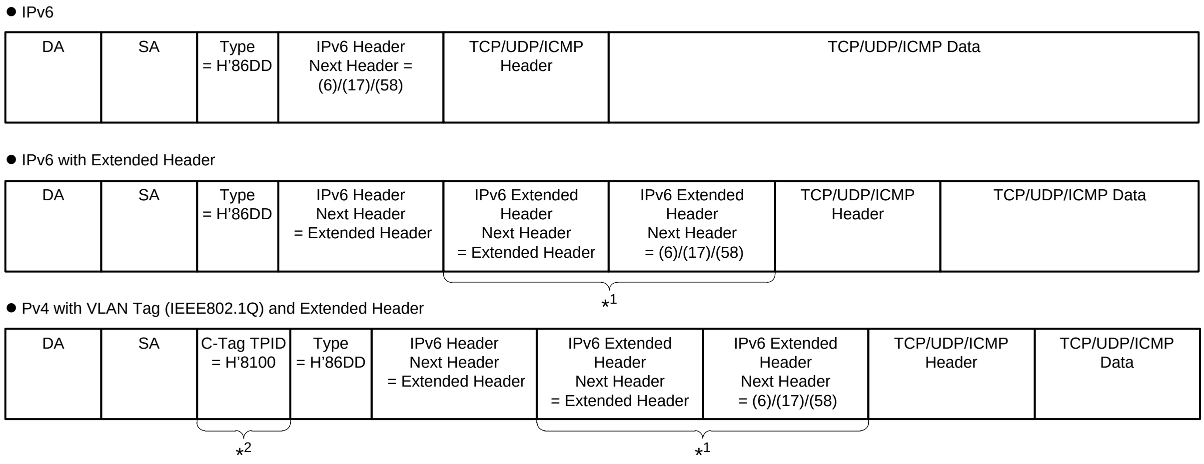


Note 1. The supported number of C-tag is only 1.
The number of 2 or more is Unsupported.

Figure 30.26 Supported Frame Format for IPv4 Checksum Calculation

(2) For IPv6 Checksum Calculation

Figure 30.27 is Supported Frame Format for the Checksum Calculation of IPv6 TCP/UDP/ICMP.



Note 1. Supported IPv6 Extended Headers (except TCP(6)/UDP(17)/ICMP(58)) are followings.

- Hop-by-Hop Options header (0)
- Routing type 0 Routing header (43)
- Destination Options header (60)
- Authentication header (51)

The Number of Supported Extended Headers is from 1 to 8.
The Number of 9 or more is Unsupported.

Note 2. The supported number of C-tag is only 1. The number of 2 or more is Unsupported.

Figure 30.27 Supported Frame Format for IPv6 Checksum Calculation

(3) Unsupported Frame Format (example)

Figure 30.28 is an Example Unsupported Frame Format.

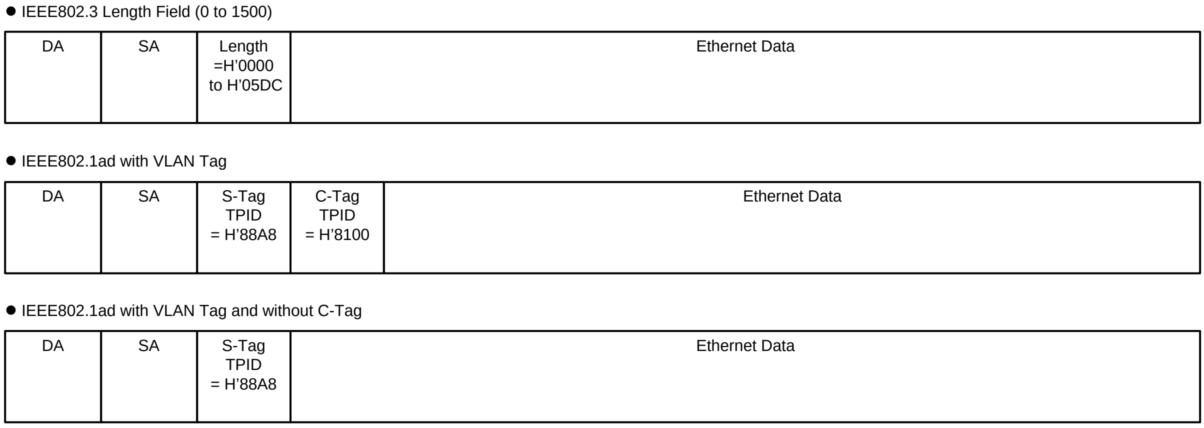


Figure 30.28 Unsupported Frame Format (example)

30.5.6.5 Checksum Calculation Flow

Figure 30.29 and **Figure 30.30** are Flows for Checksum Calculation by TOE.

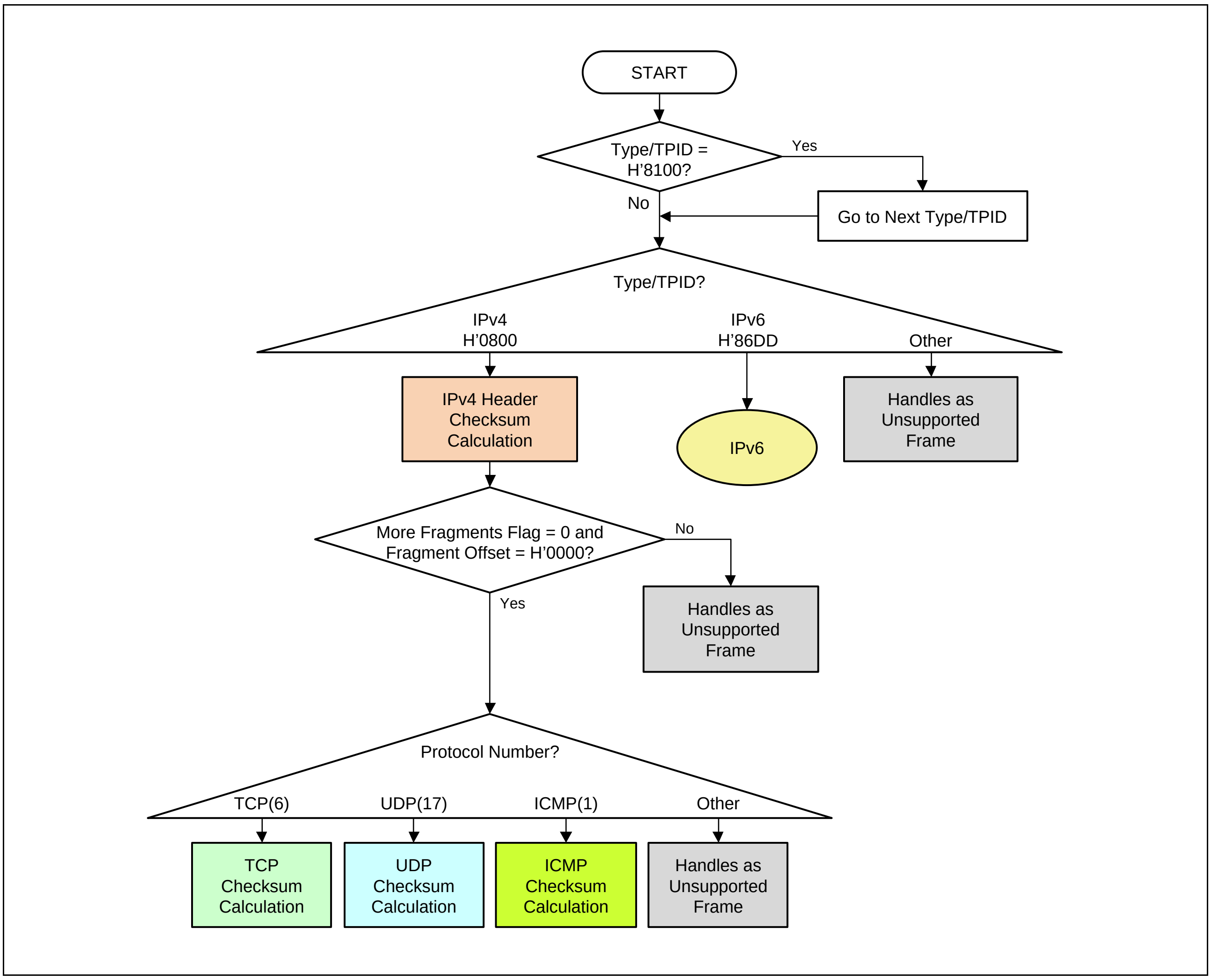


Figure 30.29 Checksum Calculation Flow (Part1)

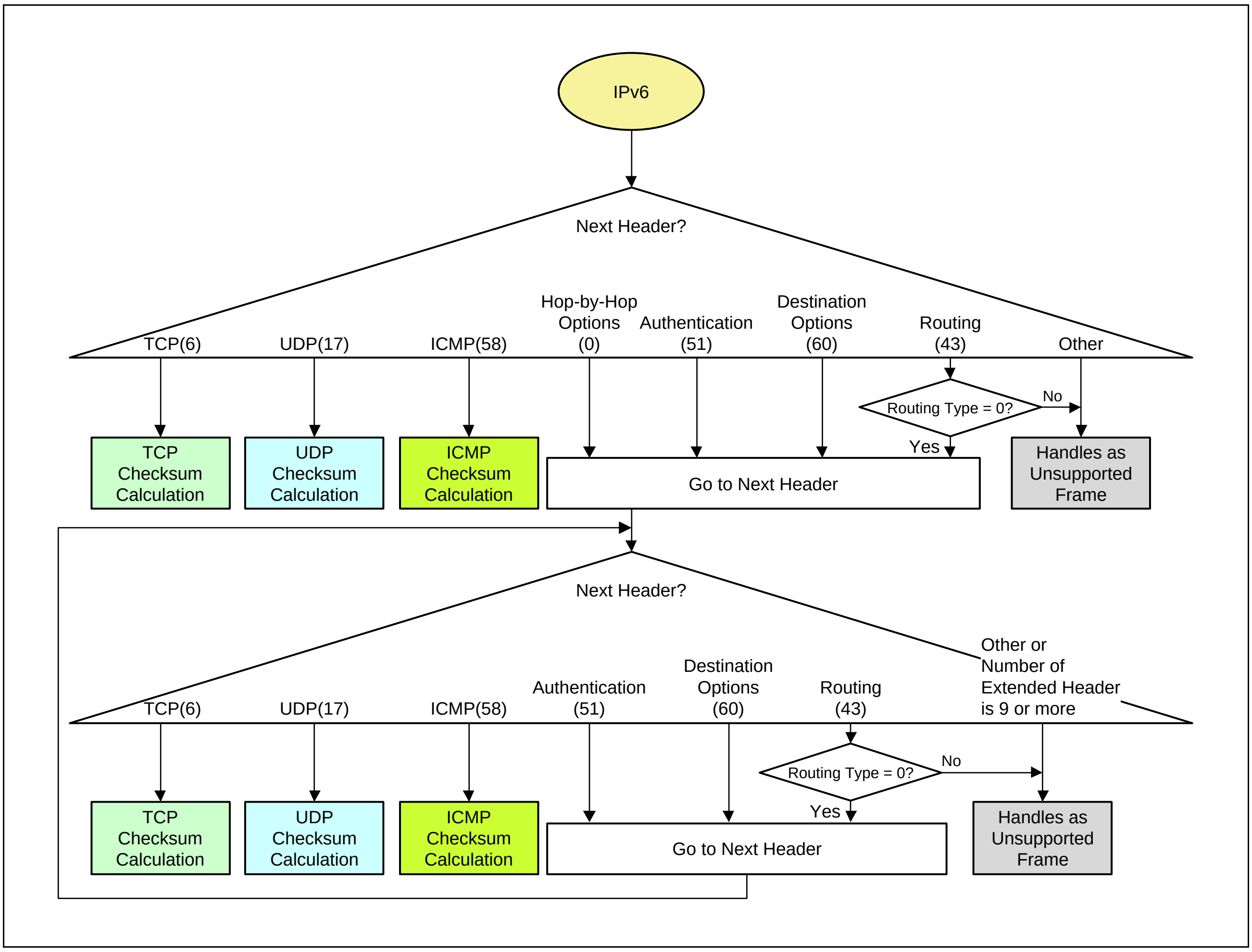


Figure 30.30 Checksum Calculation Flow (Part2)

30.5.7 Received Frame Filter

TOE has Received Frame Filters. When TOE received the frame which is not match to filters, TOE discards (abort) that frame. TOE is also able to assert an interrupt if TOE received the frame which is match to filters.

The Possible Configuration of Filter Condition are followings.

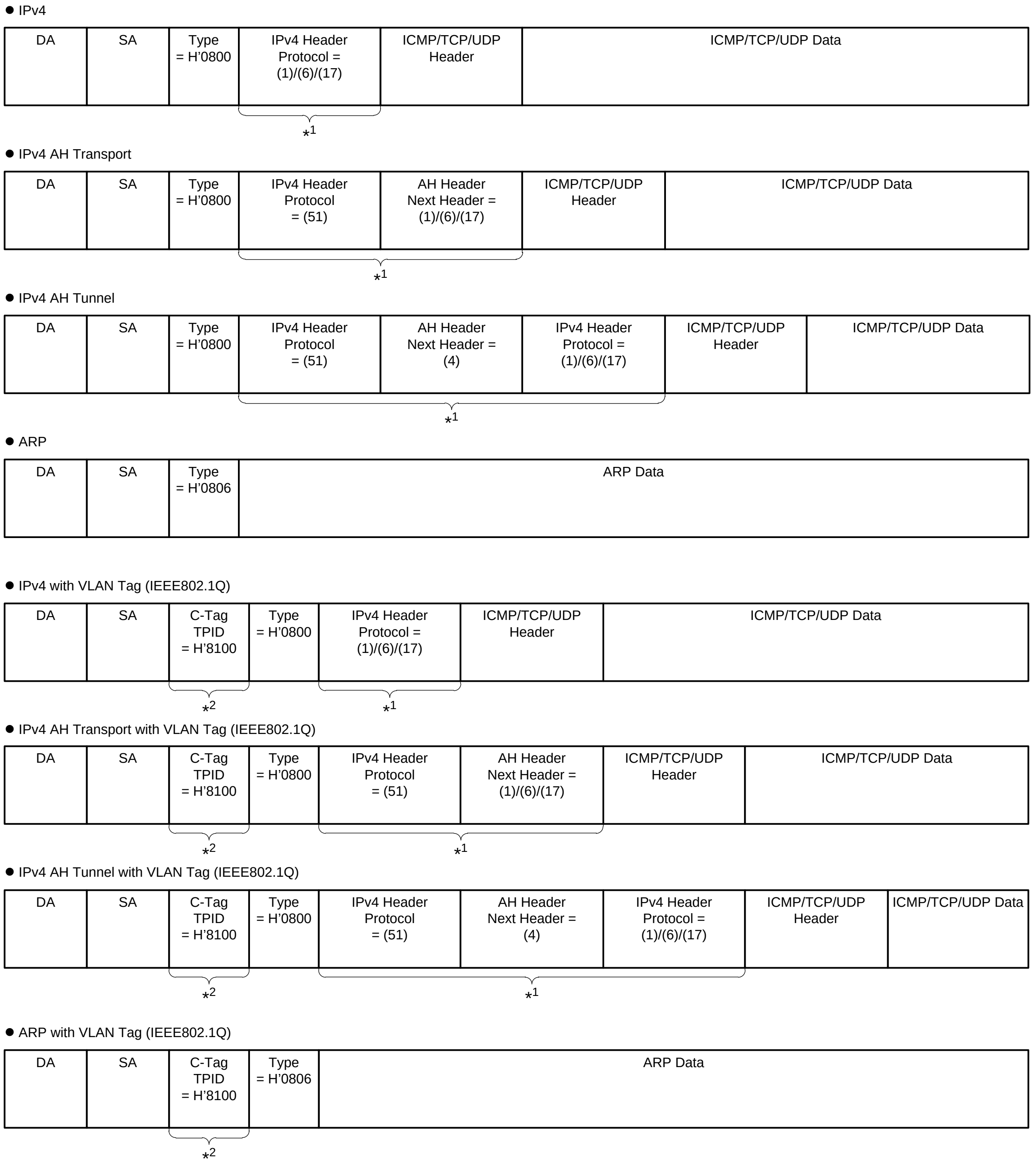
- Ethernet Type
- IP Protocol No.
- UDP destination port No.
- Destination MAC address (Unicast)
- Destination MAC address (Broadcast)
- Destination MAC address (Multicast)
- ARP Request of Local Station
- Neighbor Solicitation of Local Station
- Except IPv6 Next Header (Analyzable)

30.5.7.1 Supported Frame for Received Frame Filter

Followings are Supported /Unsupported Frame Formats by Received Frame Filter.

(1) For IPv4 Frame Filter

Figure 30.31 is Supported Frame format of IPv4.



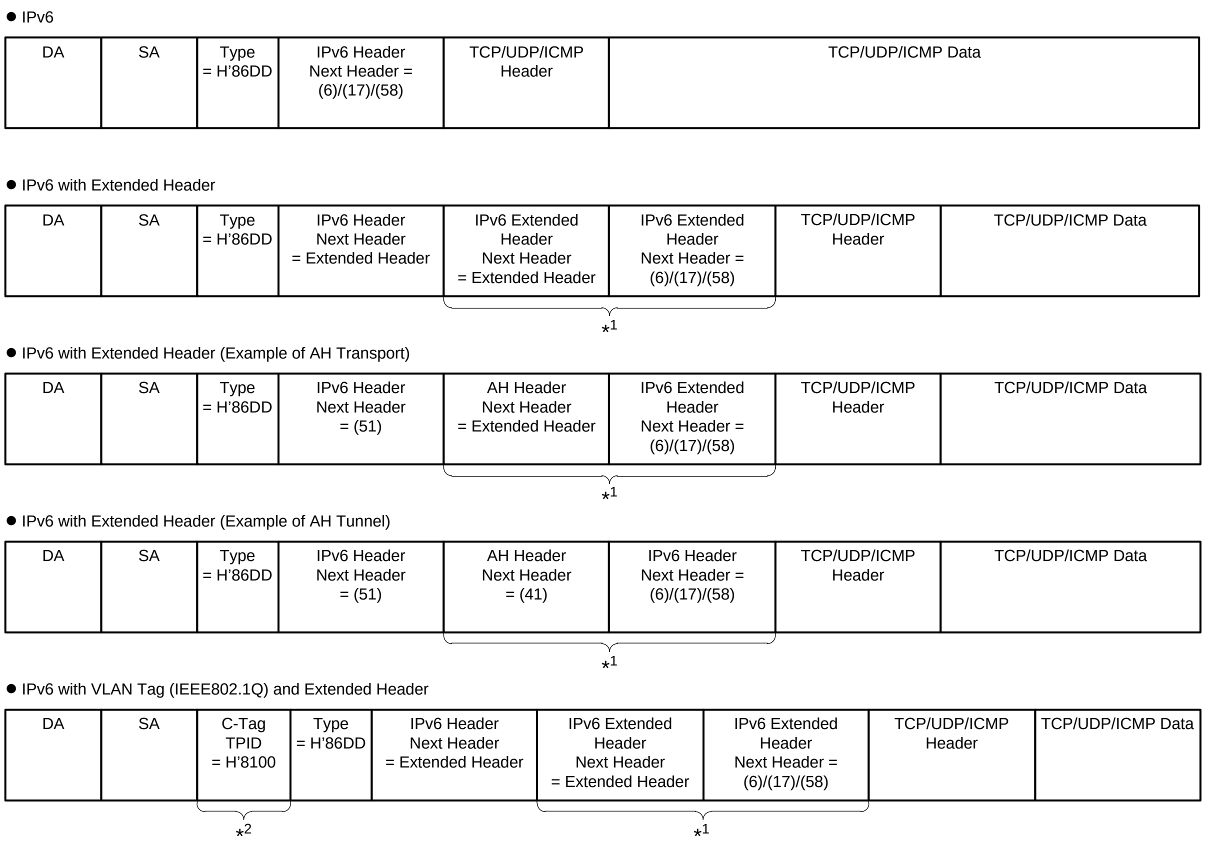
Note 1. Frame analysis is stopped if the Protocol No. is not ICMP (1), IPinIP (4), TCP (6),UDP(17) or AH(51)

Note 2. The supported number of C-tag is only 1.
The number of 2 or more is Unsupported.

Figure 30.31 Supported Frame of IPv4

(2) For IPv6 Frame Filter

Figure 30.32 is Supported Frame format of IPv6.



Note 1. Supported Extended Header (except TCP(6),UDP(17) and ICMP(58) are followings.

- Hop-by-Hop Options header (0)
- IPv6(41)
- Routing type 0 Routing header (43)
- Flag (44)
- Authentication header (51)
- Destination Options header (60)

Others are Unsupported and Frame analysis is stopped.

Note 2. The supported number of C-tag is only 1.
The number of 2 or more is Unsupported.

Figure 30.32 Supported Frame Format of IPv6

(3) Unsupported Frame Format (Example)

Figure 30.33 is Example Unsupported Frame Format.

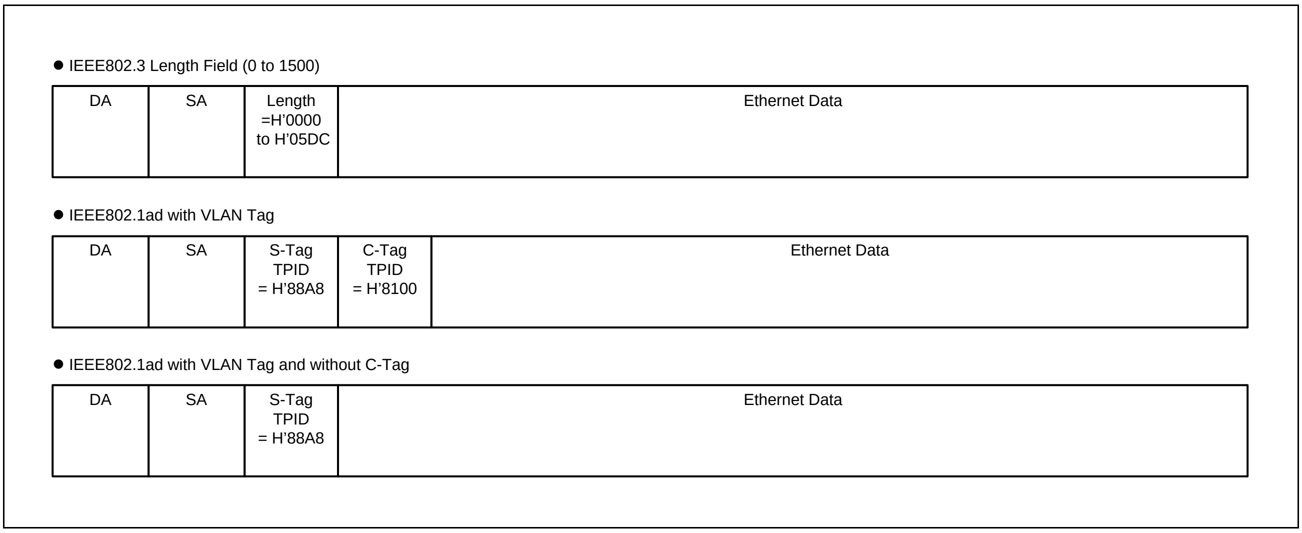


Figure 30.33 Example Unsupported Frame Format

30.5.7.2 Received Frame Filtering Flow

Figure 30.34 to **Figure 30.40** are Received Frame Filtering Flows.

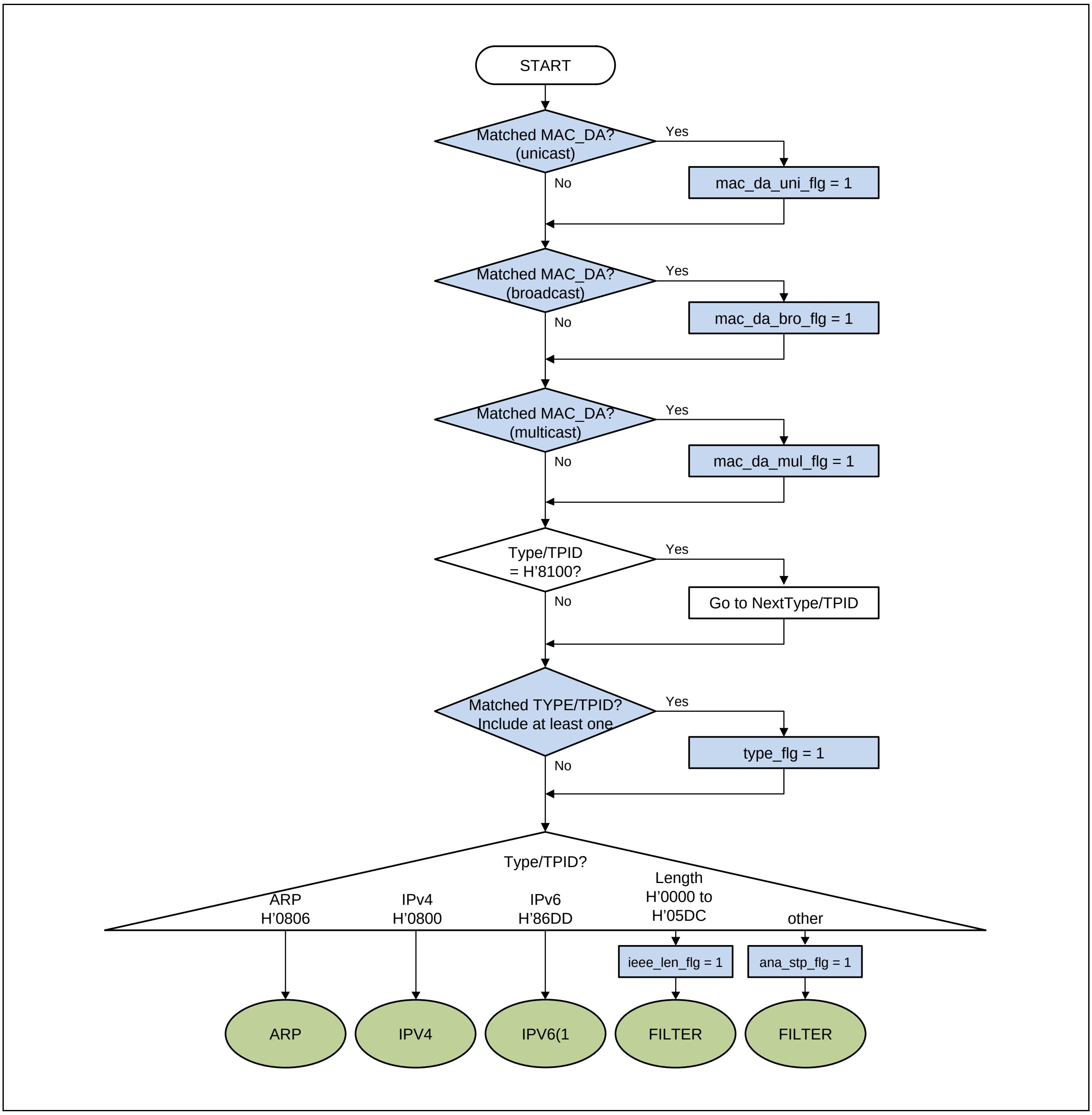


Figure 30.34 Received Frame Filtering Flow (Part1)

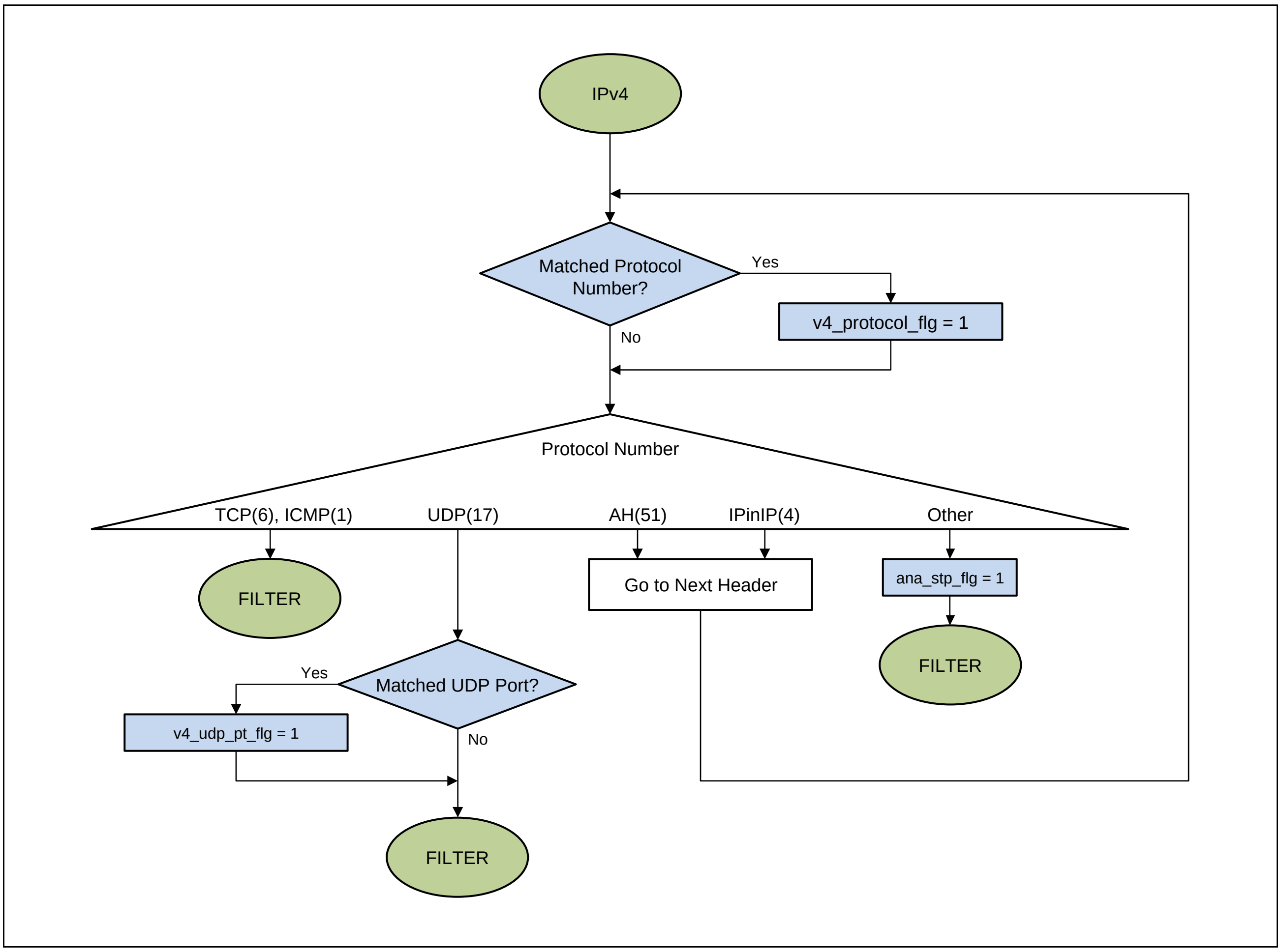


Figure 30.35 Received Frame Filtering Flow (Part2)

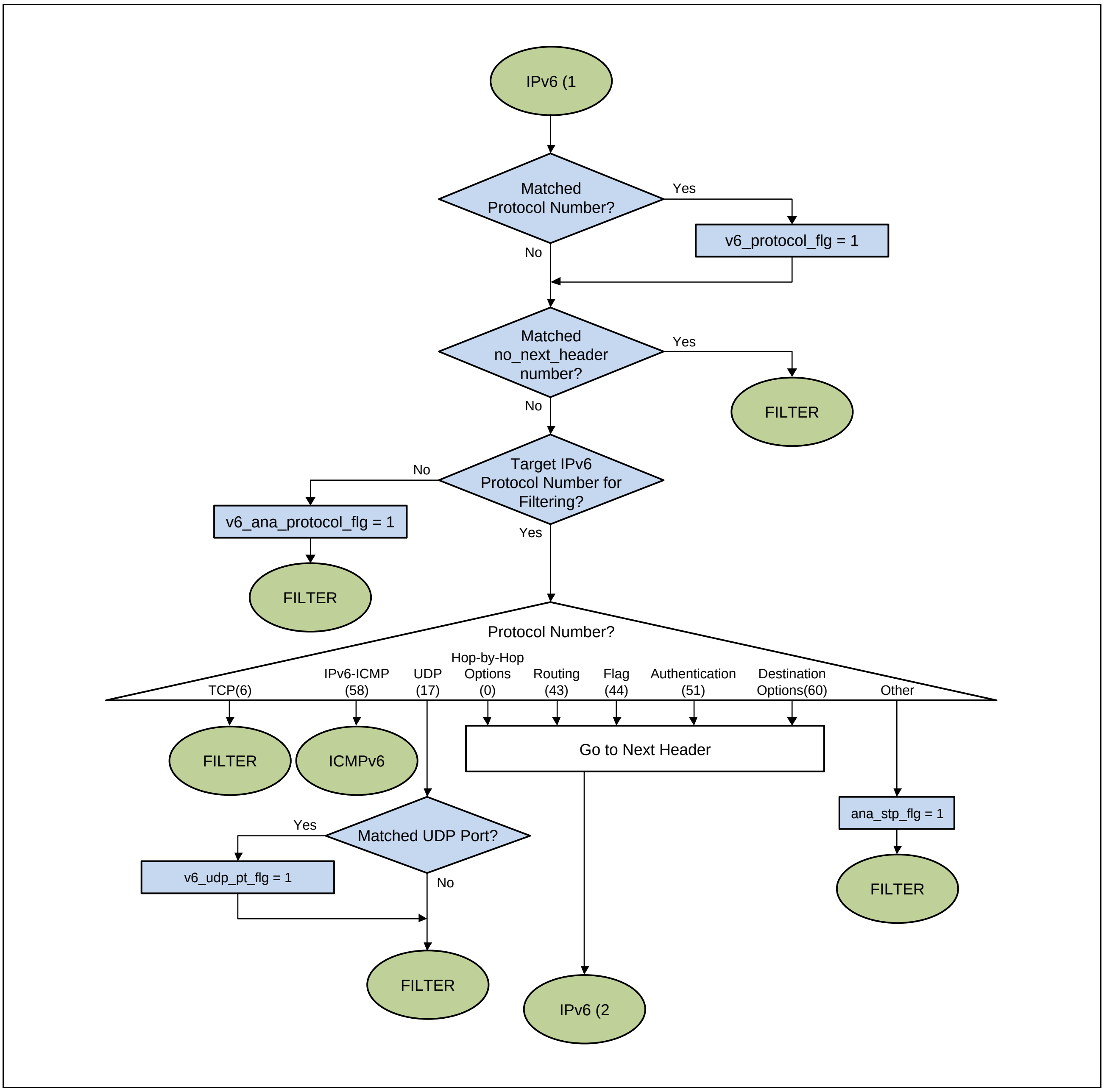


Figure 30.36 Received Frame Filtering Flow (Part3)

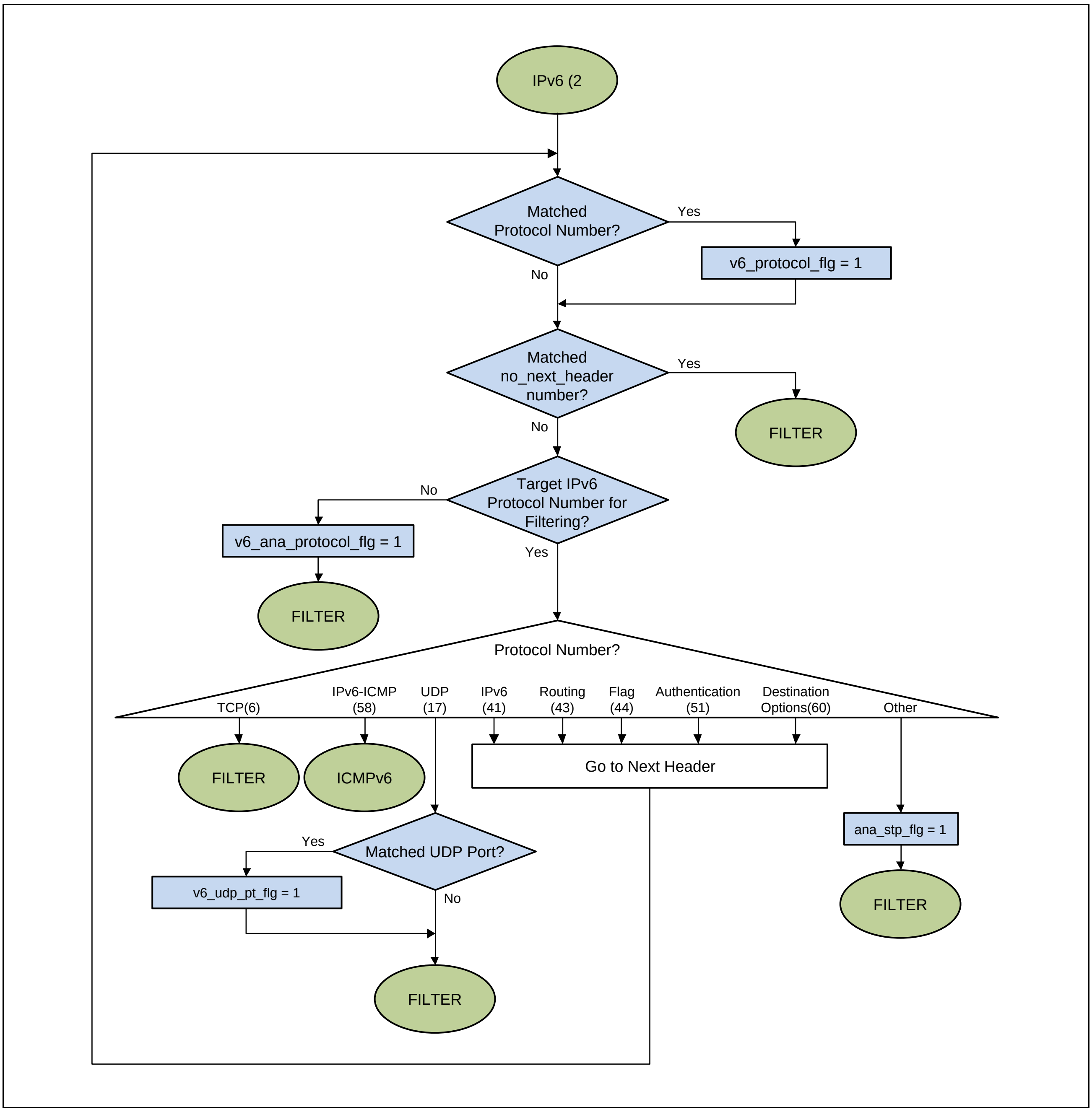


Figure 30.37 Received Frame Filtering Flow (Part4)

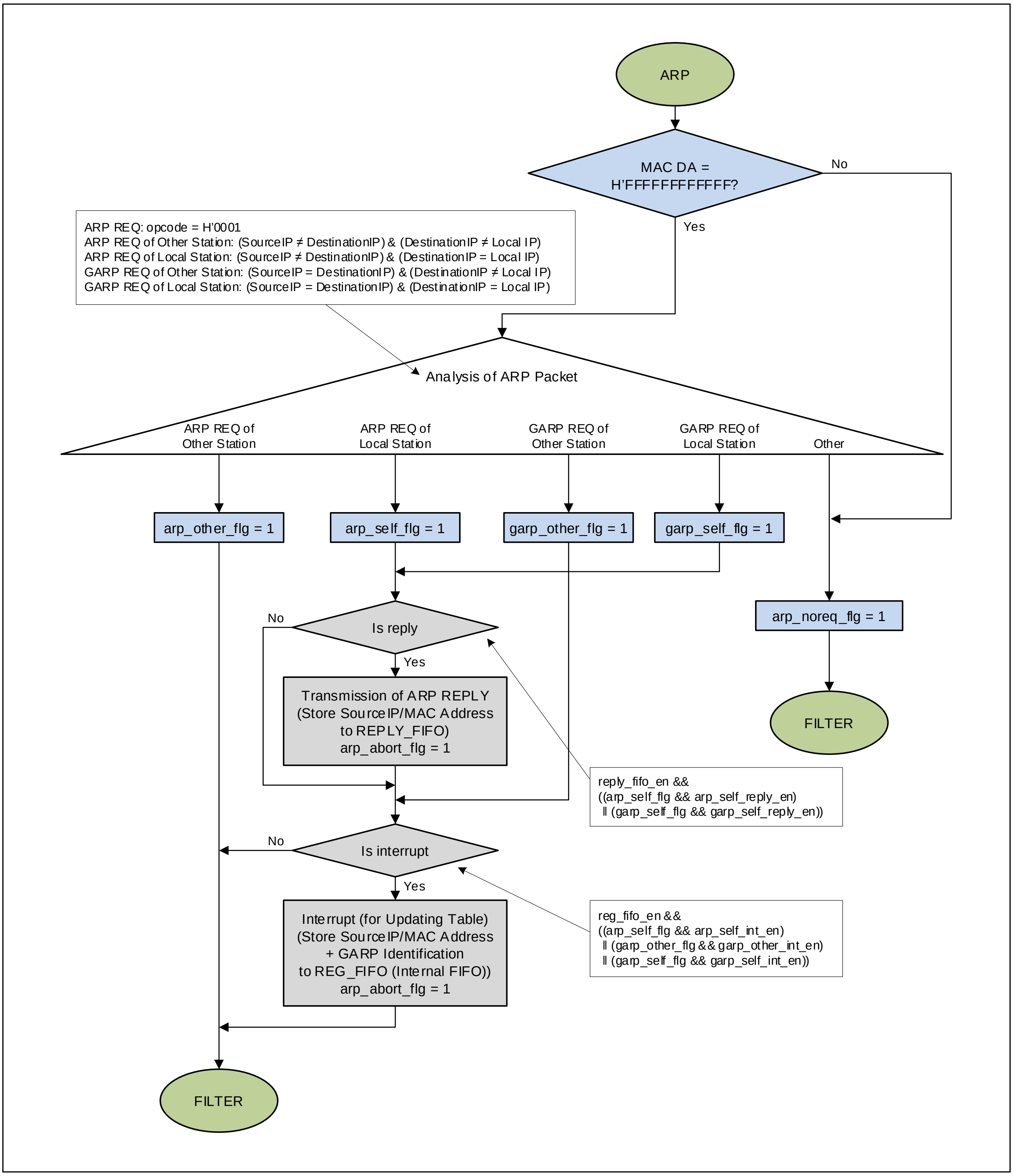


Figure 30.38 Received Frame Filtering Flow (Part5)

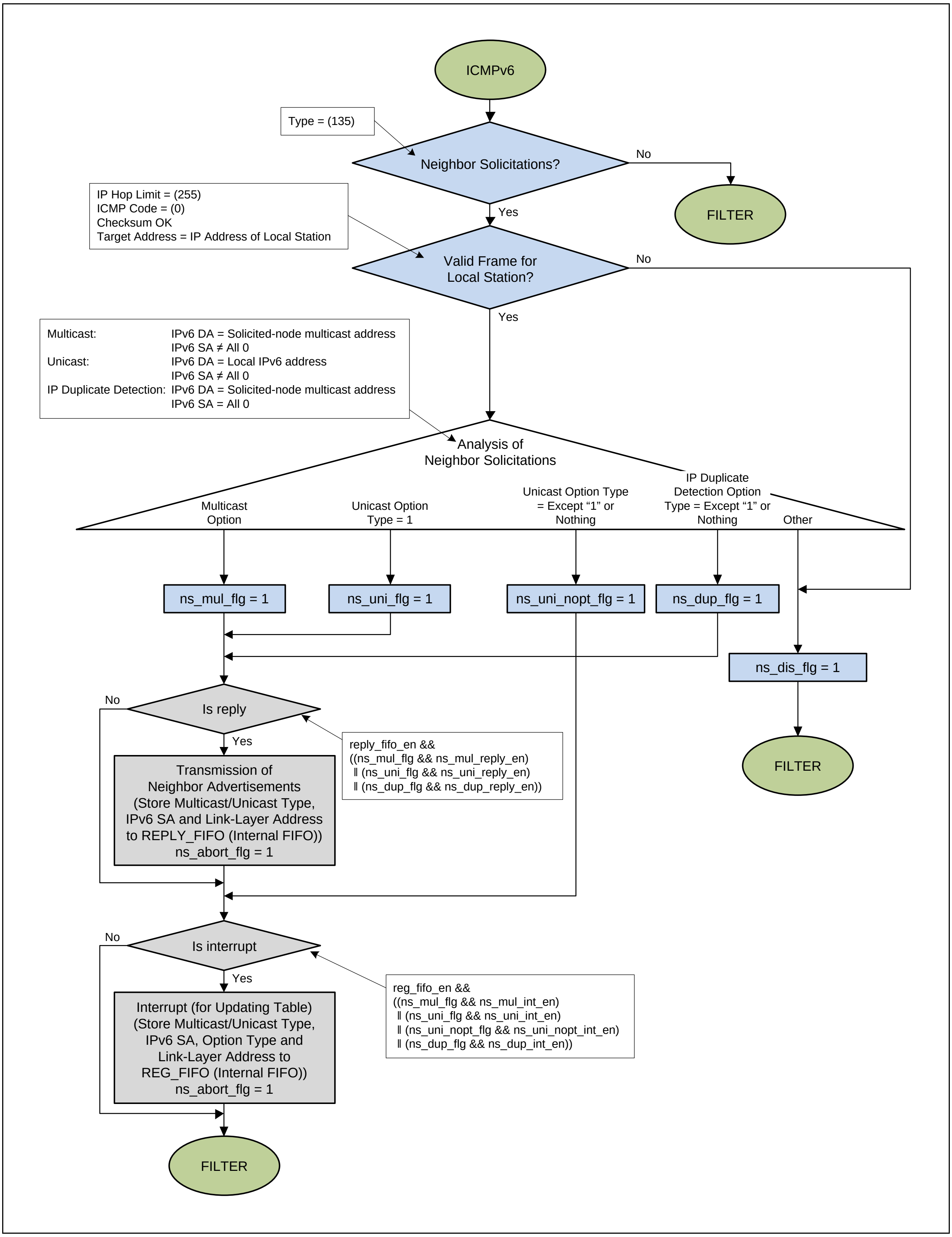


Figure 30.39 Received Frame Filtering Flow (Part6)

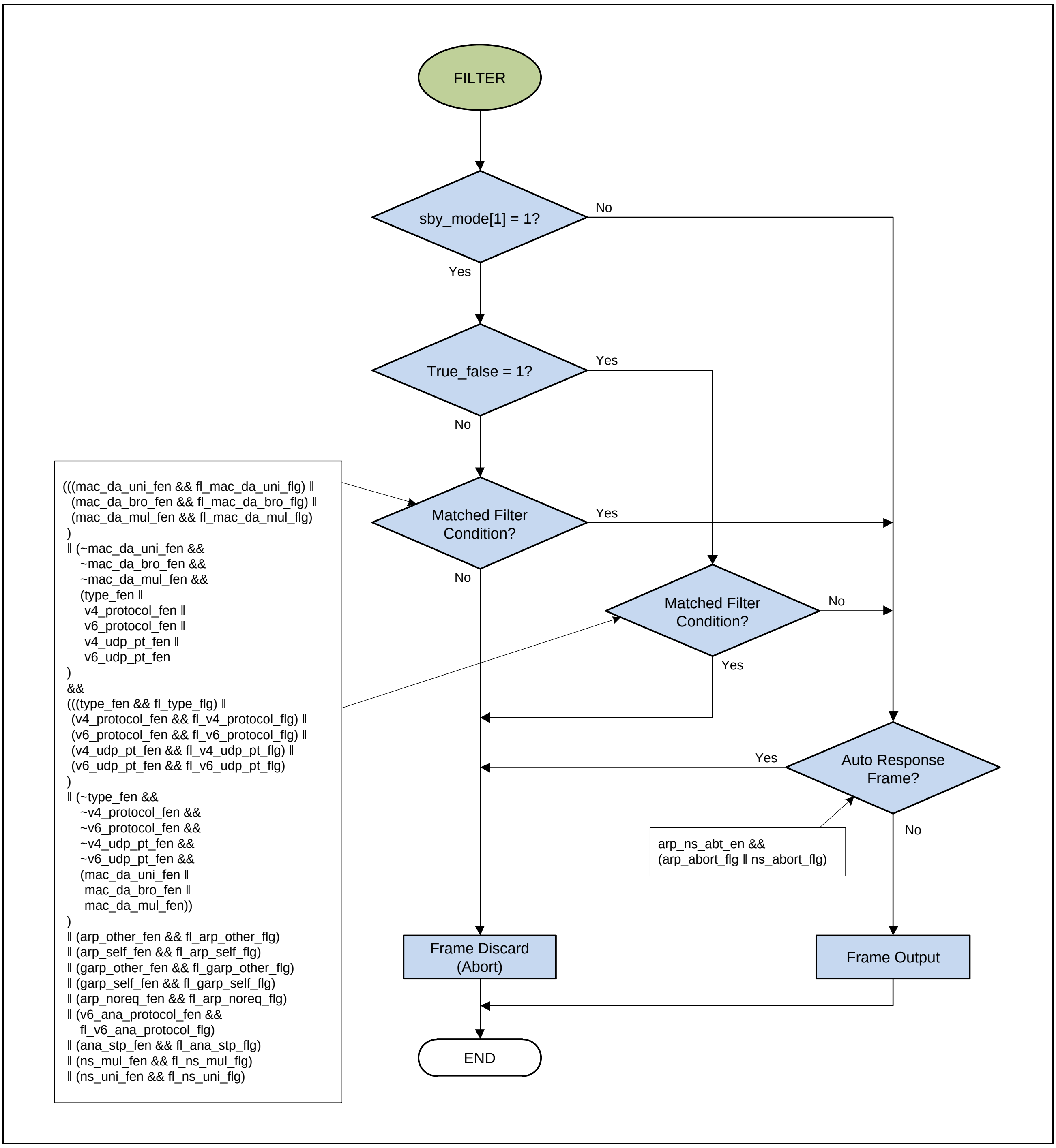


Figure 30.40 Received Frame Filtering Flow (Part7)

30.5.7.3 Filtering and Interrupts

Figure 30.41 is an overview for the function of Filtering and Interrupt.

TOE has flags for each filter conditions. When they match filter configurations of CSFR1*, each flag is set.

CSFR00 controls the action (discard or transmit) for the frames which are configured that Filtering is Enable. The filter condition relationship between “Match/Unmatch” and “Discard/Transmit” is configured by CSFR00.true_false.

The Enable/Disable configuration for discarding frame is controllable in a lump by CSFR00.sby_mode [1:0].

The flag status of each filter’s conditions is indicated to CSFR20 and outputted as interrupt. And the mask control for each interrupt is configured by CSFR21.

TOE has flags of each filter’s conditions for Filtering and Interrupt. The set condition of flags for each function is same. But clear condition is different. (The flags for Filtering are cleared at the top of frame. The flags for Interrupt are cleared by write access to the interrupt status registers.) And TOE also has Interrupt Flag which indicates the Received Frame was transmitted from TOE to DMAC without discarding by filtering conditions.

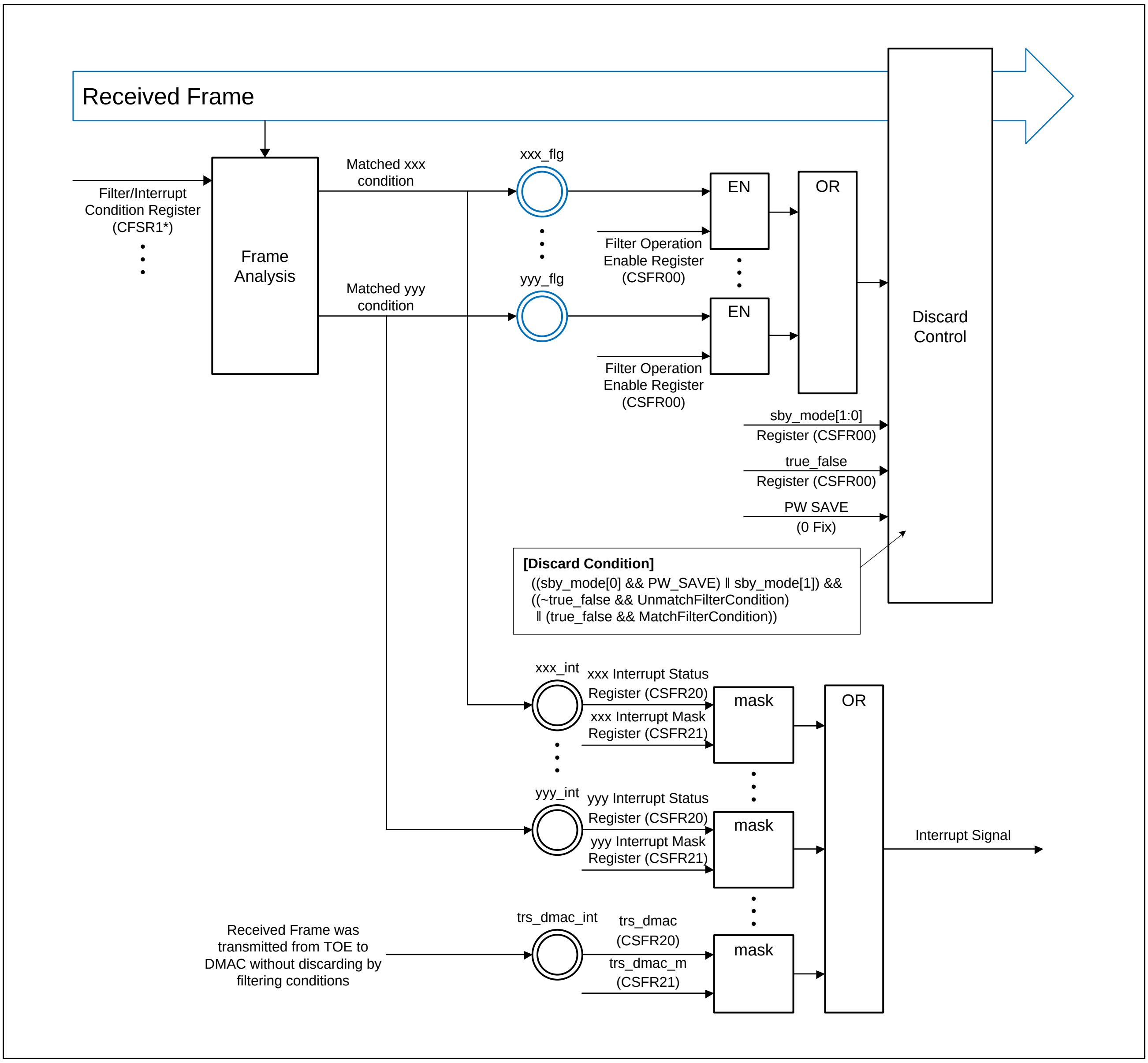


Figure 30.41 Overview of Filtering and Interrupt

30.5.8 Auto Response

TOE has following automatic responding functions.

- When TOE received ARP request for Local Station, it transmits ARP reply automatically.
- When TOE received Neighbor Solicitation for Local Station, it transmits Neighbor Advertisement automatically.

TOE also has following interrupt functions for updating ARP Table/Cache or responding function by software.

- When TOE received ARP request for Local Station, it asserts interrupt and indicates packet's information to registers. (CSFR4*)
- When TOE received Neighbor Solicitation for Local Station, it asserts interrupt and indicates packet's information to registers. (CSFR4*)

30.5.8.1 Auto Response Function for ARP REQ

Table 30.18 is operations for ARP request. See **Figure 30.38** for about the Frame Analysis Flow of ARP REQ.

Table 30.18 Operation after Received ARP REQ

	General Operation		TOE Operation		
ARP REQ	Transmit ARP REPLY	Update Table	REPLY FIFO	REG FIFO	Remarks Column
ARP for Other Station (SourceIP ≠ DestinationIP)	×	×	×	×	ARP REQ for Other Station
ARP for Local Station (SourceIP ≠ DestinationIP)	✓	✓	✓	✓	Operation of ARP for Local Station
GARP for Other Station (SourceIP = DestinationIP)	×	✓	×	✓	Request of updating ARP Table
GARP for Local Station (SourceIP = DestinationIP)	✓	×	✓	×	Response for IP Duplicate Detection

GARP:
REQ which Source IP Address and Destination IP Address are same.

REPLY FIFO:
It stores SourceIP/MAC Address to FIFO and transmits ARP REPLY by Hardware.
The information in this FIFO is cleared after TOE sends ARP REPLY.

REG FIFO:
It stores SourceIP/MAC Address + GARP identification to FIFO and asserts interrupt.
Software is expected to check registers (CSFR4*) and update ARP Table.
The information in this FIFO is cleared by writing "1" to CSRF40.arp_ns. At that time if this FIFO has next information, CSRF40.arp_ns is asserted again and CSFR4* are updated to next information.

Storing to REPLY FIFO and REG FIFO is controlled by followings.
Use followings for Enable/Disable Configuration in a lump.
CSFR00.reply_fifo_en
.reg_fifo_en

Use followings for Enable/Disable Configuration of individually

CSFR30.arp_self_reply_en
.garp_self_reply_en
.arp_self_int_en
.garp_other_int_en
.garp_self_int_en

When CSFR00.arp_ns_abt_en = “1”, the ARP frame which was stored to REPLY FIFO or REG FIFO is not transferred to DMAC. It is discarded (aborted).

REPLY FIFO and REG FIFO is used as shared FIFO for the function of “Auto Response for Neighbor Solicitation” which is indicated in the chapter of 4.8.2. And the FIFO capacity is for 8 frames.

If TOE received ARP REQ when FIFO was full, its information is not stored to FIFO and the interrupt is asserted to notify the status of by reg_fifo_full or reply_fifo_full. If one-FIFO is already full and TOE receives ARP REQ which is stored to both FIFO, its information is not stored to both FIFO.

Table 30.19 Operation after Received Neighbor Solicitation

	General Operation		TOE Operation		
Neighbor Solicitations	Transmit Neighbor Advertisements	Update Cache	REPLY FIFO	REG FIFO	Remarks Column
Multicast for Local Station Option Type = 1 (SourceIP ≠ All 0)	✓	✓	✓	✓	—
Unicast for Local Station Option Type = 1 (SourceIP ≠ All 0)	✓	✓	✓	✓	—
Unicast for Local Station Option Type ≠ 1 or Nothing (SourceIP ≠ All 0)	✓	×	×	✓	Transmission of Neighbor Advertisements is executed by software via REG_FIFO.
IP Duplicate Detection for Local Station Option Type ≠ 1 or Nothing (SourceIP = All 0)	✓	×	✓	×	Response for IP Duplicate Detection
Other	×	×	×	×	Invalid

REPLY FIFO:

It stores SourceIP/MAC Address + multicast/unicast identification to FIFO and transmits Neighbor Advertisements by Hardware.

The information in this FIFO is cleared after TOE sends Neighbor Advertisements.

REG FIFO:

It stores SourceIP/MAC Address + multicast/unicast identification + OptionType to FIFO and asserts interrupt to CPU. Software is expected handling followings by interrupt status (CSFR4*).

- Updating cache.
- In the case of Unicast for Local Station(Option Type ≠ 1 or Nothing), transmits Neighbor Advertisements after checking Source MAC Address by referencing cache or Neighbor search.

The information in this FIFO is cleared by writing “1” to CSRF40.arp_ns. At that time, if this FIFO has next information, CSRF40.arp_ns is asserted again and CSRF4* are updated to next information.

TOE does not support anycast address. If anycast address is set to CSFR02_i register, TOE behaves same action the case of received unicast address. But the Random Delay Function for responding Neighbor Advertisement which is be defined in RFC2461 is not supported.

Storing to REPLY FIFO and REG FIFO is controlled by followings.

- Use followings for Enable/Disable Configuration in a lump.
  - CSFR00.reply_fifo_en
    - .reg_fifo_en
- Use followings for Enable/Disable Configuration of individually
  - CSFR30.ns_mul_reply_en
    - .ns_uni_reply_en
    - .ns_dup_reply_en
    - .ns_mul_int_en
    - .ns_uni_int_en
    - .ns_uni_nopt_int_en
    - .ns_dup_int_en

When CSFR00.arp_ns_abt_en = “1”, the Neighbor Solicitation frame which was stored to REPLY FIFO or REG FIFO is not transferred to DMAC. It is discarded (aborted).

REPLY FIFO and REG FIFO is used as shared FIFO for the function of “Auto Response for ARP REQ” which is indicated in the chapter of 4.8.1. And the FIFO capacity is for 8 frames.

If TOE received Neighbor Solicitation when FIFO was full, its information is not stored to FIFO and the interrupt is asserted to notify the status of by reg_fifo_full or reply_fifo_full. If one-FIFO is already full and TOE receives Neighbor Solicitations which is stored to both FIFO, its information is not stored to both FIFO.

30.5.9 Flow Control

Flow control conforming to the IEEE802.3x is possible during full-duplex operation. There are two methods of transmitting PAUSE frames that are used in flow control as given below.

To use this function, configure CCC.FCE and RCR.RFCL of DMAC.

(1) Auto transmission of PAUSE frame (See bit 16 in CXR20)

A PAUSE frame is automatically transmitted by E-MAC when DMAC notifies that the condition of Reception FIFO reached at Critical Level. The value set in the CXR71 parameter settings register is used as the timer value included in the PAUSE frame. After transmitting a PAUSE frame, if the “Reception FIFO Critical Level Notification” from DMAC is not negated by the time that is indicated in the Timer value, the PAUSE frame is sent again. The number of times of transmitting a PAUSE frame can be set from 1 to 65535 in the CXR81 register, and once the upper limit of the transmission count is reached (value set in CXR81), the PAUSE frame is not transmitted thereafter.

However, if the “Reception FIFO Critical Level Notification” from DMAC is negated, the transmission count counter is reset when the next “Reception FIFO Critical Level Notification” from DMAC is asserted and PAUSE frame transmission begins. It is also possible to not define the transmission count upper limit (transmitting for infinite times).

(2) Manual transmission of PAUSE frame (See CXR72)

The PAUSE frame can be transmitted by software instruction. PAUSE frame transmission is started by setting the Timer value in the CXR72 register. In this case, PAUSE frame is transmitted once only (one frame).

(3) PAUSE Timer value (See bits 19 and 20 in CXR20)

Control can be enabled or disabled for the PAUSE frames having PAUSE Timer value 0

a) Operation during transmission
If control is enabled and the “Reception FIFO Critical Level Notification” from DMAC is negated within the time indicated by the Timer value, the PAUSE frame with Timer value 0 is transmitted.
If control is disabled, the PAUSE frame with Timer value 0 is not transmitted.

b) Operation during reception
If control is enabled, a PAUSE frame with Timer value 0 is received and frame transmission waiting is cancelled.
If control is disabled, a PAUSE frame with Timer value 0 is received and the PAUSE frame is discarded.

(4) PAUSE frame reception (See the bit 17 in CXR20)

When a PAUSE frame is received, the next frame transmission is in wait mode till the time which is considered Timer value of PAUSE frame elapses. The transmission of the frame that is being transmitted continues. In addition, the PAUSE frame reception count is counted (See CXR80).

30.5.10 Low Power Idle (LPI)

E-MAC supports the function of Low Power Idle for 1 Gbps/100 Mbps transfer rate. Since LPI function for 10Mbps is not defined in standard*[1], E-MAC also does not support.

Note 1. Specifically, Standard of 10 BASE-Te whose voltage is lower than 10 BASE-T was defined. But a basic specification except about voltage (such as the rule of Code or Media Access Control Operation) is same as 10 BASE-T. (10 BASE-Te is supported by the cable of Cat5 or more)

(1) Output function of Low Power Idle Code (LPI Code)

When E-MAC does not receive any transmission request during over certain period, E-MAC outputs transmission LPI Code to PHY for notifying request to change into Low Power Mode. If E-MAC receives a transmission request (not includes the request for PAUSE frame) while outputting transmission LPI Code, E-MAC changes output Code from transmission LPI Code to Normal Idle Code. Then E-MAC resumes to transmit Frames after waiting certain period (WakeUp time of PHY).

The following time is configurable by register.

- The time from E-MAC does not receive any transmission request to E-MAC starts outputting transmission LPI Code.
- The time from E-MAC received transmission request to E-MAC starts outputting Normal Idle Code.

(2) Receive function of Low Power Idle Code (LPI Code)

When PHY receives a changing request into Low Power Mode from Link Partner, PHY starts outputting reception LPI Code to E-MAC. If E-MAC receives reception LPI Code, E-MAC does not send this Code to TOE. E-MAC outputs Normal Status to TOE same as receiving Normal Idle Code.

(3) Stop function of transmission clock (Only RGMII Mode)

When RGMII interface is used, E-MAC can stop the RGMII transmission clock (TXC) during outputting transmission LPI Code. This function is configurable by register. But the case of MII interface is used, to stop MII transmission clock (TX_CLK) which PHY outputs is prohibited.

(4) Handling function for stopping reception clock (Both RGMII and MII Mode)

E-MAC can accept to stop RGMII and MII reception clock (RXC/RX_CLK) while receiving reception LPI Code. To stop RGMII and MII reception clock (RXC/RX_CLK) is configurable by register of PHY. (See the datasheet of PHY)

[Simplified Full Duplex Mode]

E-MAC supports Simplified Full Duplex Mode which is used for Low Power Idle Function.

In the Full Duplex Mode, when E-MAC receives transmission request from Upper Module (such as DMAC), E-MAC starts transmitting frames without care of PHY status.

But in the Simplified Full Duplex Mode, E-MAC cares the PHY status and does not transmit any frames to PHY during both PHY status is Low Power Idle and PHY is changing status from Low Power Idle to Normal.

To use E-MAC as Simplified Full Duplex Mode, set ‘1’ to DMP bit and LPM bit of CXR20 register after checking Link Partner supports Low Power Idle Mode by Auto Negotiation*[2].

Note 2. Check PHY and Link Partner support Low Power Idle Mode before setting E-MAC as Low Power Idle Mode. And for the function of Low Power Idle Mode, registers area of PHY was expanded. Therefore, to use Low Power Idle Mode, software has to be changed for the function of accessing PHY registers, PHY Auto Negotiation and initial setting of E-MAC.

30.5.11 Control of Reference Clock for Ethernet

Gigabit Ethernet Interface IP has a function to notify the status that whether Gigabit Ethernet Interface needs Reference Clock for Ethernet “clk_miitx_gtx_refclk” or not by “ref_clk_stop_en”. And this condition also is notified to CPU by interrupt signal “pif_int_n” (Case of interrupt is indicated on TIS.RCSRF and ISS.RCSRM).

This function is only enabled when the configurations of Gigabit Ethernet Interface are matched following all conditions.

- RGMII is selected as PHY IF.
- This function is enabled by register configuration. (CCC.ERCS = 1)
- E-MAC is configured as LowPowerMode. (CXR20.LPM = 1), Duplex = Full and TransmitSpeed = 1000 M/100 Mbps.

Figure 30.42 is a block diagram of this function.

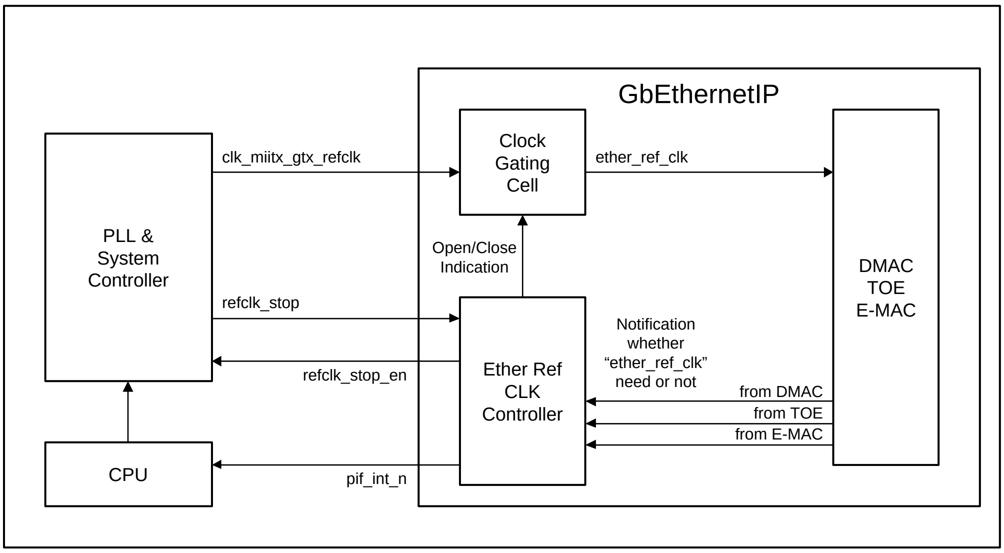


Figure 30.42 Block Diagram

As showing **Figure 30.42**, Gigabit Ethernet Interface has an “Ether Ref CLK Controller” which controls “ref_clk_stop_en”, “pif_int_n” and Open/Close Operation of Internal Clock Gating Cell for “clk_miitx_gtx_refclk”.

The state of Open/Close Operation of Internal Clock Gating Cell is indicated on CSR.RCSI.

NOTE

Whenever “pif_int_n” is asserted by this function, CPU has to instruct PLL & System Controller to stop “clk_miitx_gtx_refclk”.

30.5.11.1 Stop Procedure of Reference Clock for Ethernet

Stop procedure of “clk_miitx_gtx_refclk” is executed by following steps.

1. When DMAC and TOE do not have any Transmission Frames, they do not need “clk_miitx_gtx_refclk” anymore. So, they notify that status to Ether Ref CLK Controller.
2. After outputting Transmission LPI Code during configured time at LPTXG(M)TH4, E-MAC does not need “clk_miitx_gtx_refclk” anymore, So, it notifies that status to Ether Ref CLK Controller.
3. When Ether Ref CLK Controller is received the status that all of DMAC/TOE/E-MAC don’t need “clk_miitx_gtx_refclk” any more, it asserts “ref_clk_stop_en” to notify about this condition to ETH_PLL & System Controller. Then Gigabit Ethernet Interface closes an internal Clock Gating Cell for “clk_miitx_gtx_refclk” and asserts “pif_int_n” to indicate this condition to CPU.
4. CPU instructs ETH_PLL & System Controller to stop “clk_miitx_gtx_refclk” and writes “0” to TIS.RCSRF to clears “pif_int_n”.
5. After PLL is stopped completely, ETH_PLL & System Controller asserts “ref_clk_stop” notifying PLL status to Ether Ref CLK Controller.

Figure 30.43 is a sequence of stopping “clk_miitx_gtx_refclk”.

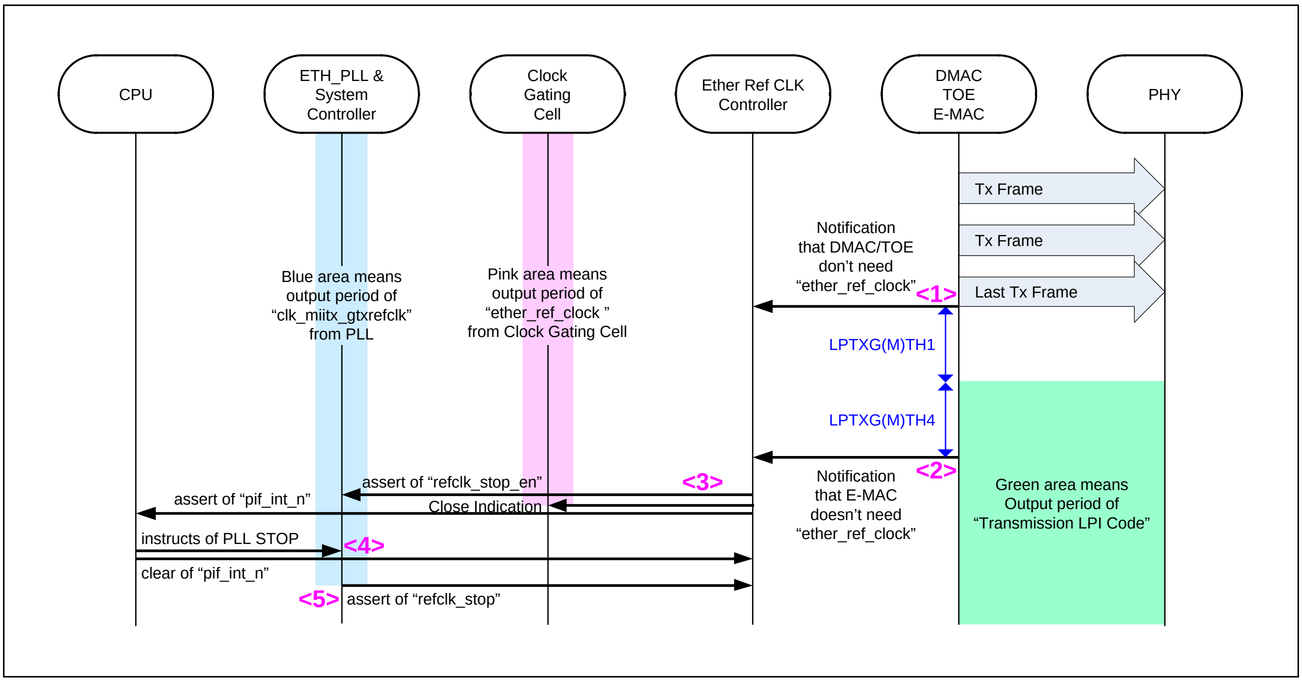


Figure 30.43 Sequence of stopping “clk_miitx_gtx_refclk”

30.5.11.2 Resume Procedure of Reference Clock for Ethernet

Resume procedure of “clk_miitx_gtx_refclk” is executed by following steps.

Case1: Gigabit Ethernet Interface receives indication from CPU restarting transmission operation.

1. CPU indicates restarting transmitting operation to DMAC.
2. DMAC fetches a new transmission descriptor and gets a Transmission Frame from external Memory. Then, it notifies to Ether Ref CLK Controller that it needs “clk_miitx_gtx_refclk” for transmitting operation.
3. After Ether Ref CLK Controller is notified from DMAC that DMAC needs “clk_miitx_gtx_refclk”, it de-asserts “ref_clk_stop_en” to notify ETH_PLL & System Controller that Gigabit Ethernet Interface needs restart of PLL.
4. ETH_PLL & System Controller tries to restart PLL. After “clk_miitx_gtx_refclk” is resumed completely, it de-asserts “ref_clk_stop” to notify Ether Ref CLK Controller about PLL status.
5. By the change of “ref_clk_stop” from assert to de-assert, Ether Ref CLK Controller opens the internal Clock Gating Cell for ”clk_miitx_gtx_refclk”.
6. Since providing clock is resumed, Transmitting Function of DMAC/TOE/E-MAC is also resumed. Then, E-MAC outputs a Transmission Frame to PHY after outputting IDLE Code during the time which is configured at LPTXG(M)TH3 register.

Figure 30.44 is a sequence of resuming “clk_miitx_gtx_refclk” of Case1.

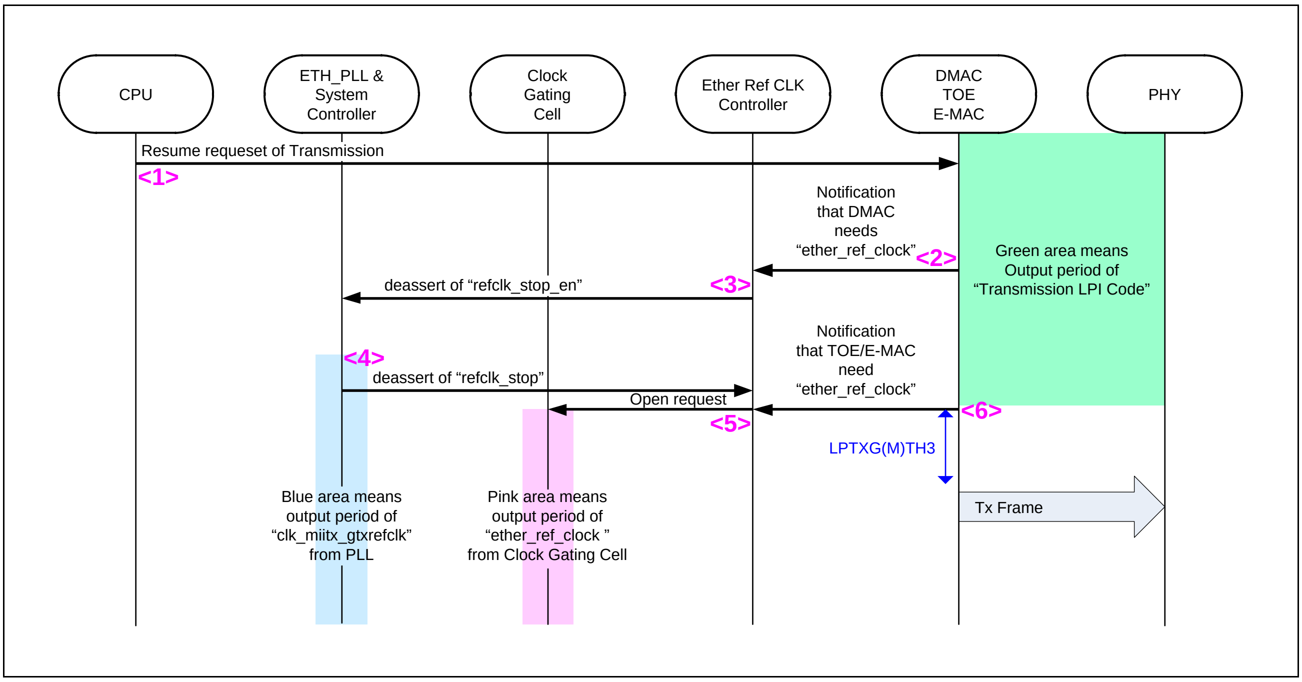


Figure 30.44 Sequence of resuming “clk_miitx_gtx_refclk” of Case1

Case2: Gigabit Ethernet Interface receives a frame which needs to Automatic Response by H/W.

For about the function of Automatic Response by H/W, see 4.8.

1. Received a frame which needs to Automatic Response by H/W from PHY.
2. After received the frame, TOE notifies to Ether Ref CLK Controller that it needs “clk_miitx_gtx_refclk” for transmitting operation.
3. After Ether Ref CLK Controller is notified from TOE that TOE needs “clk_miitx_gtx_refclk”, it deasserts “ref_clk_stop_en” to notify ETH_PLL & System Controller that Gigabit Ethernet Interface needs restart of PLL.
4. ETH_PLL & System Controller tries to restart PLL. After “clk_miitx_gtx_refclk” is resumed completely, it de-asserts “ref_clk_stop” to notify Ether Ref CLK Controller about PLL status.
5. By the change of “ref_clk_stop” from assert to de-assert, Ether Ref CLK Controller opens the internal Clock Gating Cell for ”clk_miitx_gtx_refclk”.
6. Since providing clock is resumed, Transmitting Function of DMAC/TOE/E-MAC is also resumed. Then, E-MAC outputs an Automatic Response Frame to PHY after outputting IDLE Code during the time which is configured at LPTXG(M)TH3 register.

Figure 30.45 is a sequence of resuming “clk_miitx_gtx_refclk” of Case2.

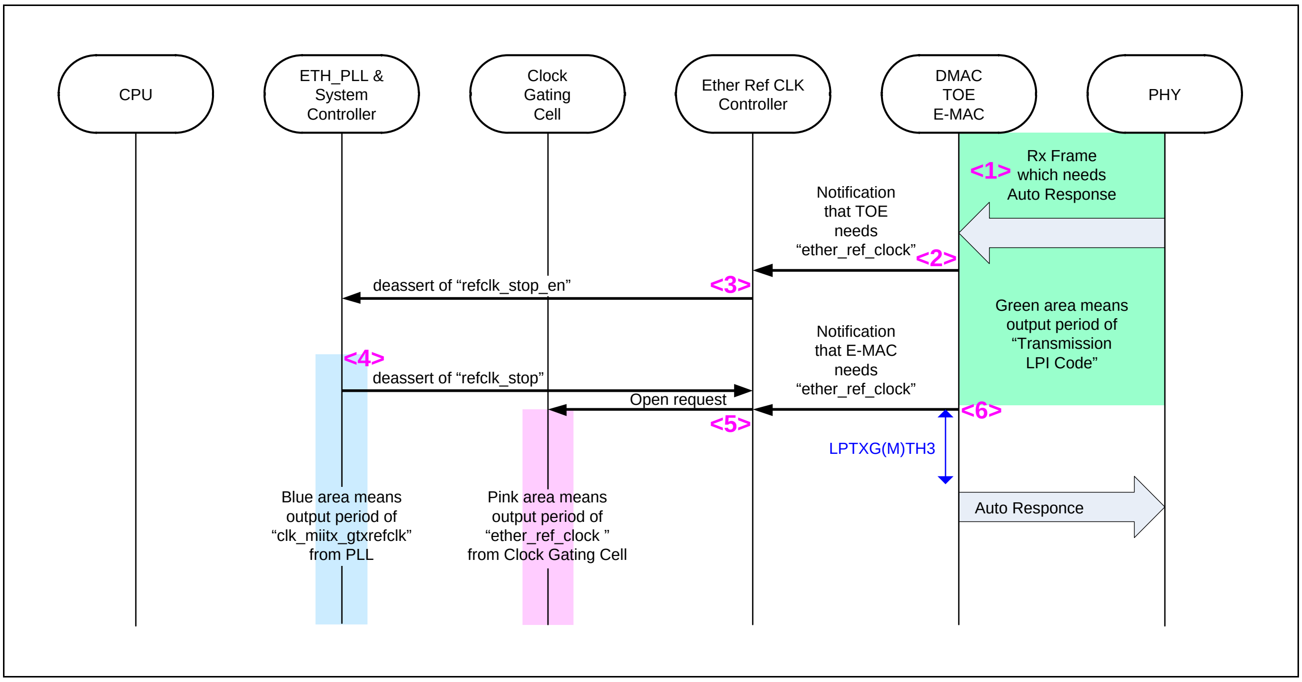


Figure 30.45 Sequence of resuming “clk_miitx_gtx_refclk” of Case2

30.5.12 Interrupts

The DMAC related interrupts include descriptor interrupts, error interrupts, reception interrupts and transmission interrupts. The states of a DMAC-related interrupt sources can be checked in the following registers.

- Descriptor interrupt status register (DIS)
- Error interrupt status register (EIS)
- Receive interrupt status register (RISi)
- Transmit interrupt status register (TIS)

The interrupts are controlled by the corresponding interrupt enable bits. However, the status flags operate independently of the settings of the enable bits.

The states of grouped interrupts can only be checked by reading the interrupt summary status register (ISS) and the queue full error interrupt summary bit in the error interrupt status register (EIS.QFS). This reduces the load on the CPU.

30.5.12.1 Transmit/Receive Data Management Interrupt

The management interrupt for transmission and reception is conveyed when the interrupt conditions corresponding to the following sources are satisfied.

- Receive frame interrupts in the receive interrupt status register 0 (RIS0.FRF)
- Descriptor processed interrupts in the receive interrupt status register 3 (RIS3.RDPF)
- Frame transmitted interrupts in the transmit interrupt status register (TIS.FTF)
- Descriptor processed interrupts in the transmit interrupt status register (TIS.TDPF)
- Descriptor processed interrupts in the descriptor interrupt status register (DIS.DPFi)

The general error interrupt state can be checked by reading the descriptor interrupt flag bits in the interrupt summary status register (ISS.DPMi) or the Reception FIFO warning interrupt summary bit (ISS.RFWM).

30.5.12.2 Error Management Interrupt

The error management interrupt is conveyed when interrupt conditions corresponding to the following sources are satisfied.

- Queue error interrupt in the error interrupt status register (EIS.QEF)
- E-MAC and TOE transmission/reception error interrupts in the error interrupt status register (EIS.MTEF, EIS.MREF, EIS.MFFF, EIS.MFFF2)
- Reception FIFO full interrupt in the receive interrupt status register 2 (RIS2.RFFF)
- Receive queue full interrupts in the receive interrupt status register 2 (RIS2.QFF0)

The general error interrupt state can be checked by reading the error interrupt summary bit in the interrupt summary status register (ISS.EM, ISS.MFWM2, ISS.MFUM2).

30.5.12.3 Other Management Interrupts

The other management interrupt is conveyed when interrupt conditions corresponding to the following sources are satisfied.

(1) Reception related interrupt

- Reception FIFO warning interrupt in the receive interrupt status register 1 (RIS1.RFWF)

(2) Transmission related interrupts

- MAC Status FIFO/FIFO2 Warning Flag (TIS.MFWF/TIS.MFWF2)
- MAC status FIFO/FIFO2 Updated Flag (TIS.MFUF/TIS.MFUF2)
- Reference Clock Stop Req Flag (TIS.RCSRF)

The general error interrupt state can be checked by reading the Reception FIFO warning error interrupt status bit in the interrupt summary status register (ISS.RFWM).

30.5.12.4 E-MAC Interrupt

The E-MAC interrupt is conveyed when the E-MAC interrupt source is generated.

- PAUSE Frame Retry Interrupt in the E-MAC status register (CXR21.PFRI)
- PHY Interrupt in the E-MAC status register (CXR21.PHYI)
- LINK Interrupt in the E-MAC status register (CXR21.LINKI)
- False Carrier Interrupt in the E-MAC status register (CXR21.FCI)

The general error interrupt state can be checked by reading the E-MAC interrupt summary bit in the interrupt summary status register (ISS.MM).

30.5.12.5 TOE Interrupt

The TOE interrupt is issued when the TOE interrupt source is generated.

- Wake Up interrupt in the TOE register (CSFR20)
- ARPREQ/Neighbor Solicitation Receive interrupt in the TOE register (CSFR40, CSFR42_*, CSFR43_*)

30.5.12.6 Interrupt

Table 30.20, **Table 30.21**, and **Table 30.22** shows interrupt for each interrupt status bit. Interrupts of DMAC and E-MAC are merged into the “pif_int_n” of output port. TOE has two independent output ports “int_fil_n” and “int_arp_ns_n”.

Table 30.20 Interrupt of DMAC

Interrupt trigger	Interrupt flag bit	Interrupt enable bit	Interrupt (Output port)
The descriptor in a reception or transmit queue has been processed where DESCR.DIE is i	DIS.DPFi (i = 1 to 15)	DIC.DPEi = 1	pif_int_n
In receive queue a frame is stored, and data is available to be processed by CPU.	RIS0.FRF	RIC0.FRE = 1	
The descriptor in reception queue has been processed where DESCR.DIE is 0001b.	RIS3.RDPF	RIC3.RDPE = 1 & CIE.CRIE = 1	
The descriptor in transmit queue has been processed where DESCR.DIE is 0001b.	TIS.TDPF	TIC.TDPE = 1 & CIE.CTIE = 1	
A frame is transmitted from transmit queue.	TIS.FTF	TIC.FTE = 1	
An error has been detected while processing receive or transmit queue.	EIS.QEF	EIC.QEE = 1	
The E-MAC has detected a fault during transmission.	EIS.MTEF	EIC.MTEE = 1	
The E-MAC has detected a fault during reception.	EIS.MREF	EIC.MREE = 1	
The MAC status FIFO2 is full.	EIS.MFFF2	EIC.MFFE2 = 1	
The MAC status FIFO is full.	EIS.MFFF	EIC.MFFE = 1	
A received frame was not completely stored in the Reception FIFO.	RIS2.RFFF	RIC2.RFFE = 1	
No sufficient space in receive queue to store completely a received frame.	RIS2.QFF0	RIC2.QFE = 1	
The Reception FIFO has reached warning level.	RIS1.RFWF	RIC1.RFWE = 1	
The MAC status FIFO has been reached warning level.	TIS.MFWF	TIC.MFWE = 1	
The MAC status FIFO has been updated.	TIS.MFUF	TIC.MFUE = 1	
The MAC status FIFO2 has been reached warning level.	TIS.MFWF2	TIC.MFWE2 = 1	
The MAC status FIFO2 has been updated.	TIS.MFUF2	TIC.MFUE2 = 1	
Reference Clock Stop Request has been notified	TIS.RCSRF	TIC.RCSRE = 1	

Table 30.21 Interrupt of E-MAC

Interrupt trigger	Interrupt flag bit	Interrupt enable bit	Interrupt (Output port)
PAUSE Frame Retry reached to threshold.	CXR21.PFRI	CXR22.PFRIM	pif_int_n
PHY Interrupt is detected.	CXR21.PHYI	CXR22.PHYIM	
LINK status change is detected.	CXR21.LINKI	CXR22.LINKIM	
False Carrier is detected.	CXR21.FCI	CXR22.FCIM	

Table 30.22 Interrupt of TOE

Interrupt trigger	Interrupt flag bit	Interrupt mask bit	Interrupt main enable bit	Interrupt (Output port)
Transmit Rx Frame to DMAC	CSFR20.trs_dmac	CSFR21. trs_dmac_m	—	int_fil_n
Reception of IEEE Length Frames.	CSFR20.ieee_len	CSFR21. ieee_len_m	—	
Reception of Invalid Neighbor Solicitations.	CSFR20.ns_dis	CSFR21.ns_dis_m	—	
Reception of IP Duplicate Detection Neighbor Solicitations.	CSFR20.ns_dup	CSFR21.ns_dup_m	—	
Reception of unicast Neighbor Solicitations. (Except “Option Type=1” or No “Option Type”)	CSFR20.ns_uni_nopt	CSFR21.ns_uni_nopt_m	—	
Reception of unicast Neighbor Solicitations. (“Option Type=1”)	CSFR20.ns_uni	CSFR21.ns_uni_m	—	
Reception of multicast Neighbor Solicitations.	CSFR20.ns_mul	CSFR21.ns_mul_m	—	
Reception of Un-analyzable Frames.	CSFR20.ana_stp	CSFR21.ana_stp_m	—	
Reception of Un-analyzable IPv6 Protocol No.	CSFR20.v6_ana_protocol	CSFR21.v6_ana_protocol_m	—	
Reception of ARP (Except ARP REQ).	CSFR20.arp_noreq	CSFR21.arp_noreq_m	—	
Reception of GARP REQ for Local Station.	CSFR20.garp_self	CSFR21.garp_self_m	—	
Reception of GARP REQ for Other Station.	CSFR20.garp_other	CSFR21.garp_other_m	—	
Reception of ARP REQ for Local Station.	CSFR20.arp_self	CSFR21.arp_self_m	—	
Reception of ARP REQ for Other Station.	CSFR20.arp_other	CSFR21.arp_other_m	—	
Match of IPv6 UDP Port No.	CSFR20.v6_udp_pt	CSFR21.v6_udp_pt_m	—	
Match of IPv4 UDP Port No.	CSFR20.v4_udp_pt	CSFR21.v4_udp_pt_m	—	
Match of IPv6 Protocol No.	CSFR20.v6_protocol	CSFR21.v6_protocol_m	—	
Match of IPv4 Protocol No.	CSFR20.v4_protocol	CSFR21.v4_protocol_m	—	
Match of Ether Type.	CSFR20.type	CSFR21.type_m	—	
Match of Multicast MAC DA.	CSFR20.mac_da_mul	CSFR21.mac_da_mul_m	—	
Match of Broadcast MAC DA.	CSFR20.mac_da_bro	CSFR21.mac_da_bro_m	—	
Match of Unicast MAC DA.	CSFR20.mac_da_uni	CSFR21.mac_da_uni_m	—	
REPLY_FIFO is full.	CSFR40.reply_fifo_full	CSFR41.reply_fifo_full_m	CSFR00.reply_fifo enable & at least one CSFR30.xx_reply_en	int_arp_ns_n
REG_FIFO is full.	CSFR40.reg_fifo_full	CSFR41.reg_fifo_full_m	CSFR00.reg_fifo_en & at least one CSFR30.xx_int_en	
Reception of ARPREQ or Neighbor Solicitations and Stored to REG FIFO.	CSFR40.arp_ns	CSFR41.arp_ns_m	CSFR00.reg_fifo_en & at least one CSFR30.xx_int_en	

30.5.13 Configuration Procedure

The following are Configuration Procedures. The meaning of “X” in Write Value, set the appropriate value to each bit for your system.

30.5.13.1 Set Up Procedure

Check the Configuration of Other Station by using Auto Negotiation Function of PHY before setting Transfer Rate (1 Gbps/100 Mbps/10 Mbps), Duplex Mode (Half or Full), Low Power Mode and Flow Control to E-MAC registers.

Action		Address	Write Value	Description
Hardware Reset				
↓				
DMAC.CSR	Read	H’00C	—	Check the read value is same as “H’00000001” (Reset Mode).
↓				
DMAC.CCC	Write	H’000	H’00000001	Set “Config Mode”
↓				
DMAC.CSR	Read	H’00C	—	Check the read value is same as “H’00000002” (Config Mode).
↓				
E-MAC Initial Configuration (Not necessary to change other E-MAC registers)				
E-MAC.CXR35	Write	H’540	H’XXXXXXXX	Set the appropriate value for your system. And wait the double of your configuration value to this register.
E-MAC.CXR31	Write	H’530	H’0000000X	Set the appropriate value for your system. (Only if In-Band Status of RGMII will be used)
E-MAC.CXR2C	Write	H’52C	H’0000000X	Set the appropriate value for your system.
E-MAC.CXR2A	Write	H’508	H’00002000	Set Max Frame size as 8KB.
E-MAC.CXR71	Write	H’554	H’0000XXXX	Set the appropriate value for your system. (Only if Flow Control Function will be used)
E-MAC.CXR24	Write	H’5C0	H’XXXXXXXX	Set the appropriate value for your system.
E-MAC.CXR25	Write	H’5C8	H’0000XXXX	Set the appropriate value for your system.
E-MAC.CXR21	Write	H’510	H’0000001F	Clear all Interrupt.
E-MAC.CXR22	Write	H’518	H’0000000X	Set the appropriate value for your system.
E-MAC.CXR2D	Write	H’5B0	H’000000X0	Set the appropriate value for your system. (Only if In-Band Status of RGMII will not be used)
E-MAC.CXR2G	Write	H’5BC	H’0000000X	Set the appropriate value for your system. (Only if Software LINK will not be used)
↓				
TOE Initial Configuration (Not necessary to change other TOE registers).				
TOE.CSR1	Write	H’804	H’0XX000XX	Set the appropriate value for your system.
TOE.CSR2	Write	H’808	H’0XX000XX	Set the appropriate value for your system.
TOE.CSFR01	Write	H’844	H’XXXXXXXX	Set the appropriate value for your system.
TOE.CSFR02_i	Write	H’848+4*i	H’XXXXXXXX	Set the appropriate value for your system. (i = 0 to 3)
TOE.CSFR03_U	Write	H’858	H’0000XXXX	Set the appropriate value for your system.
TOE.CSFR03_L	Write	H’85C	H’XXXXXXXX	Set the appropriate value for your system.
TOE.CSFR10_i	Write	H’870+4*i	H’XXXXXXXX	Set the appropriate value for your system. (i = 0 to 3)
TOE.CSFR10	Write	H’880	H’000000XX	Set the appropriate value for your system.
TOE.CSFR11_i	Write	H’884+4*i	H’XXXXXXXX	Set the appropriate value for your system. (i = 0 to 3)
TOE.CSFR11	Write	H’894	H’00000XXXX	Set the appropriate value for your system.
TOE.CSFR12_i	Write	H’898+4*i	H’XXXXXXXX	Set the appropriate value for your system. (i = 0 to 11)
TOE.CSFR12	Write	H’8C8	H’00XXXXXX	Set the appropriate value for your system.
TOE.CSFR13_U	Write	H’8CC	H’0000XXXX	Set the appropriate value for your system.
TOE.CSFR13_L	Write	H’8D0	H’XXXXXXXX	Set the appropriate value for your system.
TOE.CSFR14_U	Write	H’8D4	H’0000XXXX	Set the appropriate value for your system.
TOE.CSFR14_L	Write	H’8D8	H’XXXXXXXX	Set the appropriate value for your system.
TOE.CSFR15_U_i	Write	H’8DC+8*i	H’0000XXXX	Set the appropriate value for your system. (i = 0 to 19)
TOE.CSFR15_L_i	Write	H’8E0+8*i	H’XXXXXXXX	Set the appropriate value for your system. (i = 0 to 19)
TOE.CSFR15	Write	H’97C	H’XX0XXXXX	Set the appropriate value for your system.
TOE.CSFR20	Write	H’9A0	H’000FFFFF	Clear all Interrupt.
TOE.CSFR21	Write	H’9A4	H’000XXXXX	Set the appropriate value for your system.

Action		Address	Write Value	Description
TOE.CSFR30	Write	H'9B0	H'000XXXXX	Set the appropriate value for your system.
TOE.CSFR31	Write	H'9B4	H'XX0XXXXX	Set the appropriate value for your system.
TOE.CSFR00	Write	H'840	H'10XXXXXX	Enable the appropriate Filter Function for your system. Disable of Auto Response Function.
↓				
DMAC Initial Configuration (Not necessary to change other DMAC registers).				
DMAC.DBAT	Write	H'004	H'XXXXXXXX	Set the appropriate value for your system.
DMAC.RCR	Write	H'090	H'XXXX000X	Set the appropriate value for your system.
DMAC.RTC	Write	H'0B4	H'00002000	Set Max Reception Frame size as 8 KB.
DMAC.CIE	Write	H'384	H'00000X0X	Set the appropriate value for your system.
DMAC.DIE	Write	H'450	H'0000XXXX	Set the appropriate value for your system.
DMAC.EIE	Write	H'458	H'00000X0X	Set the appropriate value for your system.
DMAC.RIE0	Write	H'460	H'0000000X	Set the appropriate value for your system.
DMAC.RIE1	Write	H'468	H'X0000000	Set the appropriate value for your system.
DMAC.RIE2	Write	H'470	H'X000000X	Set the appropriate value for your system.
DMAC.TIE	Write	H'478	H'000XXX0X	Set the appropriate value for your system.
DMAC.RIE3	Write	H'488	H'0000000X	Set the appropriate value for your system.
↓				
DMAC.CCC	Write	H'000	H'0X0X0002	Set bit [1:0] = "10b" (Operation Mode) and bit25 (FCE)/bit18 (ERCS) = "the appropriate value for your system".
↓				
TOE.CSR0	Write	H'800	H'00000030	Set TPE and RPE = "1" and TOE FIFO=2 KB.
↓				
E-MAC.CXR20	Write	H'500	H'0XXXXXXX	Set the appropriate value for your system. (The following are example)
			H'02030061	Rx CRC PassThrough, PAUSE, TPE and RPE = '1', Half Duplex (for 100 Mbps/10 Mbps), Promiscuous.
			H'02030063	Rx CRC PassThrough, PAUSE, TPE/RPE = '1', Full Duplex (for 10 Mbps), Promiscuous.
			H'02030073	Rx CRC PassThrough, PAUSE, TPE/RPE = '1', Simplified Full Duplex (for 1 Gbps/100 Mbps), Promiscuous.
↓				
DMAC.TCCR	Write	H'304	H'000X0001	Set bit0 (TSRQ) = "1" and bit18 (MFEN2), bit16 (MFEN) = "the appropriate value to for your system".

30.5.13.2 System Mode Change Procedure (Normal -> Network Standby)

Action		Address	Write Value	Description
DMAC.TCCR	Read	H'304	—	Wait until bit0 (TSRQ) will be "0".
↓				
DMAC.CSR	Read	H'00C	—	Wait until bit16 (TPO) and bit20 (RPO) will be "0".
↓				
DMAC.CCC	Write	H'000	H'0X0X0202	Set bit9 (RDFD) = "1" to stop fetch of Rx Descriptor. Not necessary to change other bits.
↓				
DMAC.CSR	Read	H'00C		Wait until bit9 (RDFDM) will be "1" and re-check bit20 (RPO) is "0".*1
↓				
TOE Configuration to enable Auto Response Function (Not necessary to change other registers)				
TOE.CSFR40	Write	H'9C0	H'00000007	Clear all Interrupt.
TOE.CSFR41	Write	H'9C4	H'0000000X	Set the appropriate value for your system.
TOE.CSFR00	Write	H'840	H'1XXXXXXX	Enable of the appropriate Auto Response Function for your system. Not necessary to change Filter Function.
↓				
Change System Mode from "Normal" to "Network Standby".				
↓				
Set Tx Descriptor and Tx Frame at Internal SRAM				
↓				
DMAC.DBAT	Write	H'004	H'XXXXXXXX	Set the appropriate value for your system.
↓				
DMAC.DLR	Write	H'008	H'003FFFFF	Set bit0 and bit4 = "1" loading DBAT to Tx and Rx queue.
↓				

Action		Address	Write Value	Description
DMAC.TCCR	Write	H'304	H'000X0001	Set bit0 (TSRQ) = "1" and bit18 (MFEN2), bit16 (MFEN) = "the appropriate value to for your system".

Note 1. If bit9 (RDFDM) is "0" and bit20 (RPO) is "0", wait bit9 (RDFDM) will be "1" without accessing other registers.
If bit9 (RDFDM) is "1" and bit20 (RPO) is "0", the DMA transfer and fetch of Rx Descriptor have stopped.
So go to next step.
If bit9 (RDFDM) is "0" and bit20 (RPO) is "1", wait bit9 (RDFDM) will be "1" without accessing other registers.
If bit9 (RDFDM) is "1" and bit20 (RPO) is "1", the Reception Frame was stored to Reception FIFO.
So software has to cancel changing System mode. Set DMAC.CCC. bit9 (RDFD) = "0" immediately to resume fetch of Rx Descriptor.

30.5.13.3 System Mode Change Procedure (Network Standby -> Normal)

Action		Address	Write Value	Description
TOE.CSFR00	Write	H'840	H'10XXXXXX	Disable of the Auto Response Function. Not necessary to change Filter Function.
↓				
DMAC.TCCR	Read	H'304	—	Wait until bit0 (TSRQ) will be "0".
↓				
DMAC.CSR	Read	H'00C	—	Wait until bit16 (TPO) will be "0".
↓				
Change System Mode from "Network Standby" to "Normal".				
↓				
Set Rx/Tx Descriptor and Tx Frame at External SDRAM.				
↓				
DMAC.DBAT	Write	H'004	H'XXXXXXXX	Set the appropriate value for your system.
↓				
DMAC.DLR	Write	H'008	H'003FFFFF	load DBAT to Tx and Rx queue.
↓				
DMAC.CCC	Write	H'000	H'0X0X0002	Set "bit9 (RDFD) = 0" to start Rx DMA Transfer for Rx Frame. Not necessary to change other bits.
↓				
DMAC.TCCR	Write	H'304	H'000X0001	Set bit0 (TSRQ) = "1" and bit18 (MFEN2), bit16 (MFEN) = "the appropriate value to for your system".

30.5.13.4 Stop Procedure

Action		Address	Write Value	Description
Make “EOS” in the Tx Descriptor chain and wait until it is used. (Wait the interrupt of DIS.DPFi of this EOS Descriptor.)				
↓				
E-MAC.CXR20	Write	H’500	H’0XXXXXXX	Set bit6 (RPE) = “0” to stop Rx Function of E-MAC. Not necessary to change other bits.
↓				
Wait following time depending on the Transfer speed mode. (Waiting time that the last Rx Frame is stored to Reception FIFO and the last Tx Frame is transmitted to PHY) • 1 Gbps : Wait 70 µs • 100 Mbps : Wait 700 µs • 10 Mbps : Wait 7 ms				
↓				
DMAC.CSR	Read	H’00C	—	Wait until bit20 (RPO) will be “0” (Wait Reception FIFO will be empty)
↓				
DMAC.CCC	Write	H’000	H’0X0X0102	Set bit8 (DTSR) = “1” to stop URAM access. Not necessary to change other bits.
↓				
DMAC.CSR	Read	H’00C	—	Wait until bit8 (DTS) will be “1” (URAM access is stopping)
↓				
TOE.CSR0	Write	H’800	H’00000000	Set bit5 (RPE) = “0” and bit4 (TPE) = “0” to stop Rx and Tx Function of TOE. Not necessary to change other bits.
↓				
TOE.CSR0	Read	H’800	—	Wait until bit5 (RPE) and bit4 (TPE) will be “0”
↓				
E-MAC.CXR20	Write	H’500	H’0XXXXXXX	Set bit5 (TPE) = “0” to stop Tx Function of E-MAC. Not necessary to change other bits.
↓				
E-MAC.CXR20	Read	H’500	—	Wait until bit6 (RPE) and bit5 (TPE) will be “0”
↓				
DMAC.CCC	Write	H’000	H’0X0X0001	Set bit [1:0] = “01b” (Config Mode) and bit8 (DTSR) = “0”

Note: If the system needs software reset, set DMAC.CCC bit [1:0] = “00b” to be RESET Mode after “Stop Procedure” and retry “Set up Procedure”.

30.5.14 Connection to PHY-LSI

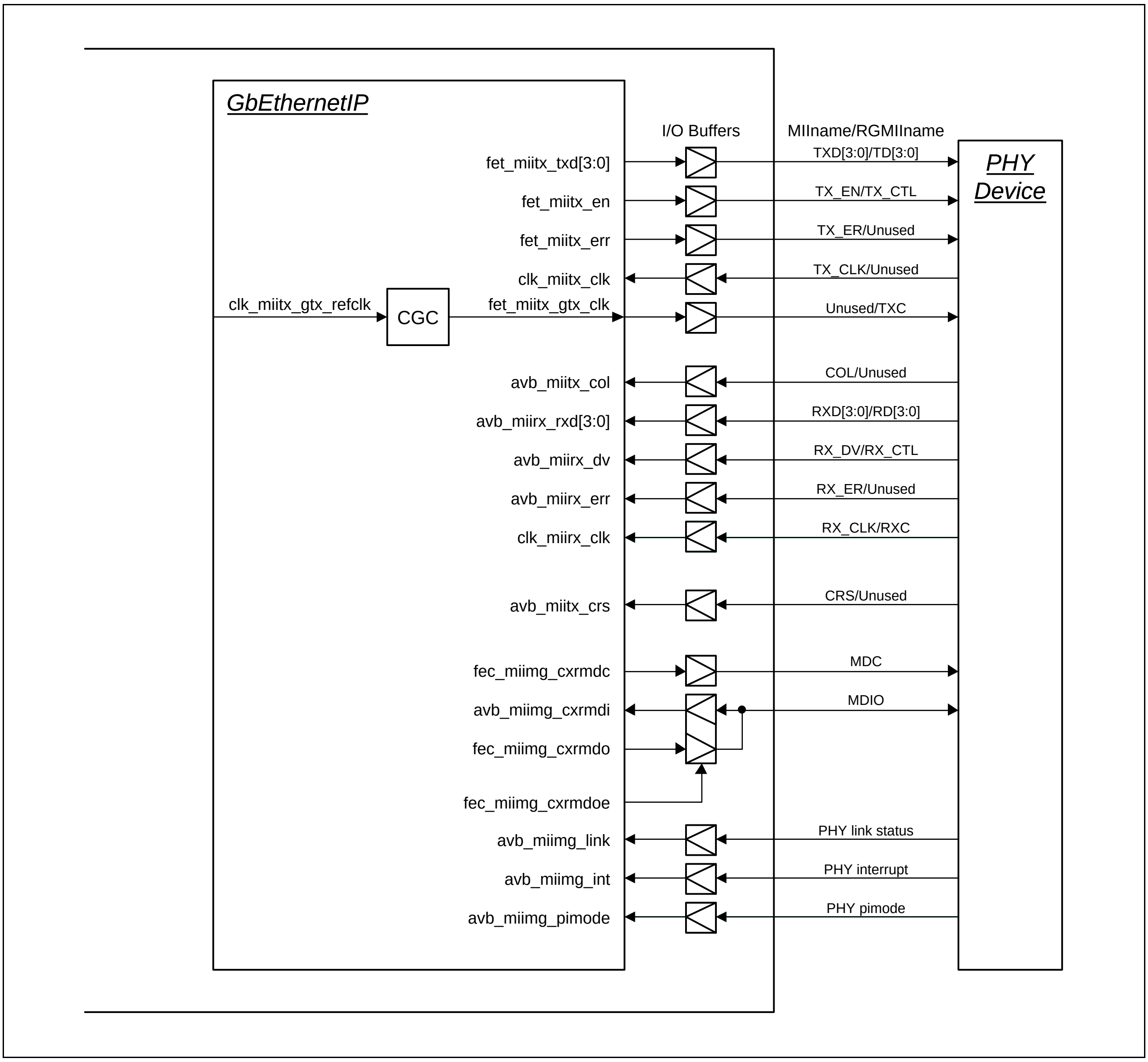


Figure 30.46 Connection to PHY-LSI

30.5.14.1 RGMII Frame Transmission/Reception Timing

Each RGMII frame transmission/reception timing is shown in **Figure 30.47** and **Figure 30.48**.

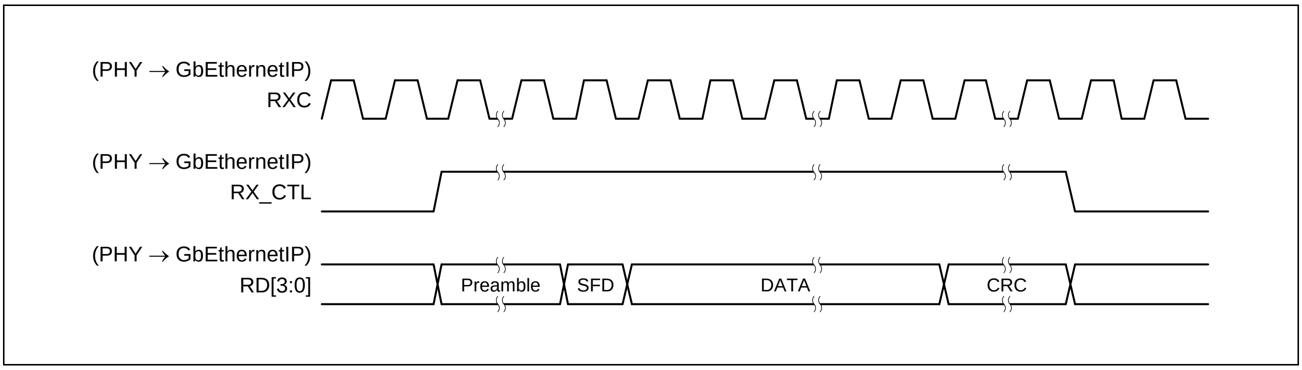


Figure 30.47 Normal Frame Reception Time Chart (RGMII)

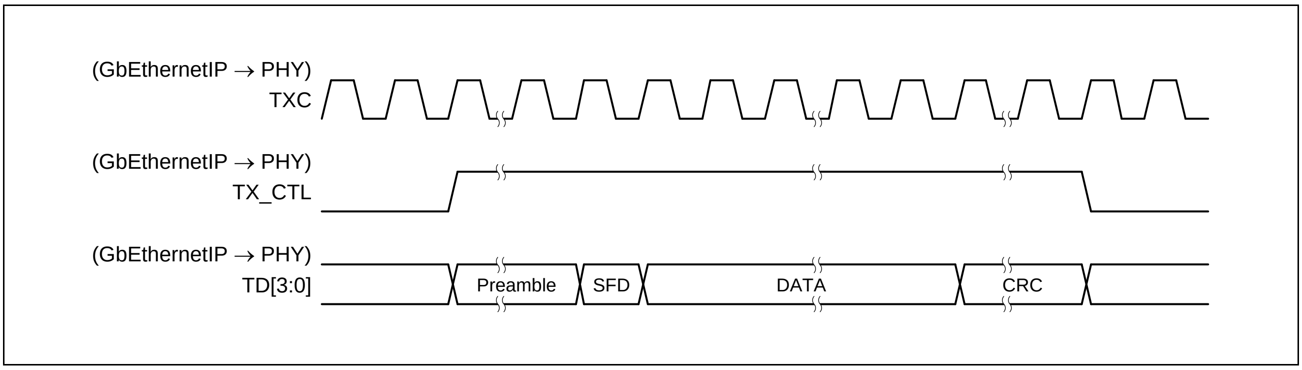


Figure 30.48 Normal Frame Transmission Time Chart (RGMII)

30.5.14.2 MII Frame Reception Timing

Each MII frame transmission/reception timing is shown in **Figure 30.49** and **Figure 30.50**.

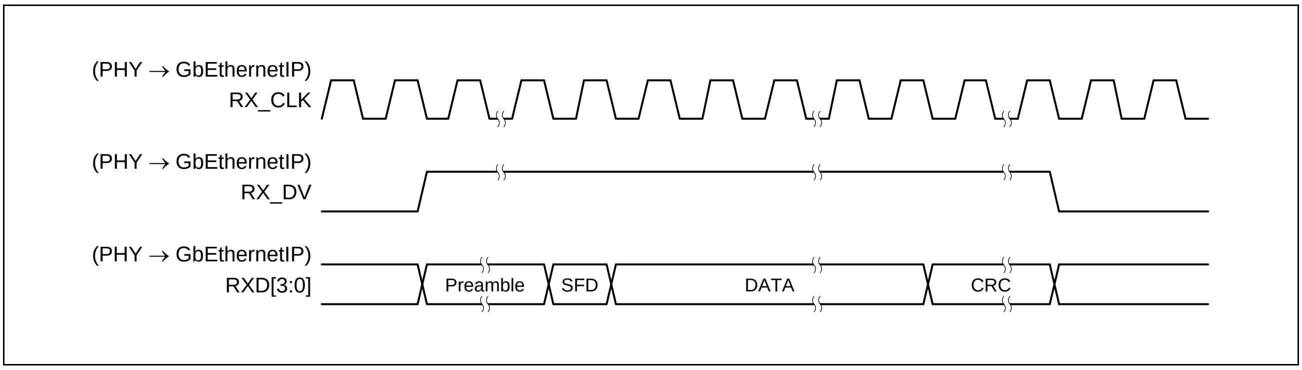


Figure 30.49 Normal Frame Reception Time Chart (MII)

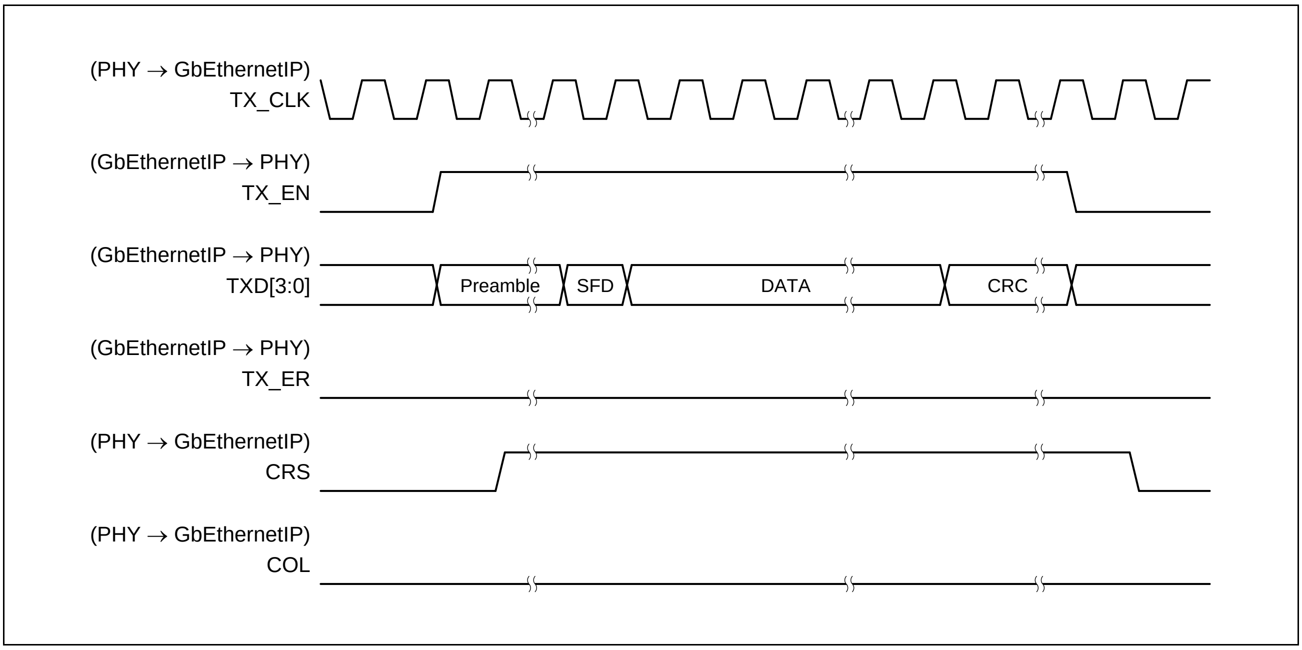


Figure 30.50 Normal Frame Transmission Time Chart (MII)

30.5.14.3 Accessing PHY Registers

There are two methods to access the PHY register: using the E-MAC-MDIOC block and using the CXR23 register.

Which access method to be used can be set using the MDIOMOD register.

(1) PHY Register Access Method (using MDIOC)

To access the PHY register from MDIOC, the SEL_MDIO bit in MDIOMOD register must be set to 0.

The following commands can be used for indirect access using MDIOC.

Table 30.23 List of MDIOC Commands

Command	OP code	Address	Data	Description
WRITE	“01”	A [9:0]	D [15:0]	Write PHY register
READ	“10”	A [9:0]	D [15:0]	Read PHY register

PHY Register Write Access Procedure (Using MDIOC)

The write access procedure for PHY register is given below.

(1) Read MDIOSTS register and confirm that BSY bit = 0.

(2) Set PHY address as (A [9:5]) and register address as (A [4:0]) in MDIOADR register.

(3) Set write data in MDIODAT register.

(4) Set OP field of MDIOCMD register to “01”.
{Here, the hardware sets BSY bit in MDIOSTS register to 1.}

(5) Read MDIOSTS register and wait till BSY bit = 0. After BSY becomes 0, write access is complete.
{After write access to PHY register completes, the BSY bit in MDIOSTS register is cleared to 0 by hardware.}

PHY Register Read Access Procedure (Using MDIOC)

The read access procedure for PHY Register is given below.

(1) Read MDIOSTS register and confirm that BSY bit = 0.

(2) Set PHY address as (A [9:5]) and register address as (A [4:0]) in MDIOADR register.

(3) Set OP field of MDIOCMD register to “10”.
{Here, the hardware sets BSY bit in MDIOSTS register to 1.}

(4) Read MDIOSTS register and wait till BSY bit = 0.
{After read access from PHY register completes and read data becomes available in MDIODAT register, hardware clears BSY bit in MDIOSTS register is cleared to 0 by hardware.}

(5) When the BSY bit in MDIOSTS register is cleared to 0, read data is acquired from MDIODAT register and access is complete.

(2) PHY Register Access Method (Using CXR23)

To access PHY register using CXR23 register, SEL_MDIO bit of MDIOMOD register must be set to 1. (The initial value can be used as is.)

PHY access using CXR23 is done by controlling High and Low of MDC and MDIO signals for every cycle. The relationship between CXR23 register access and MDC and MDIO signals is shown below.

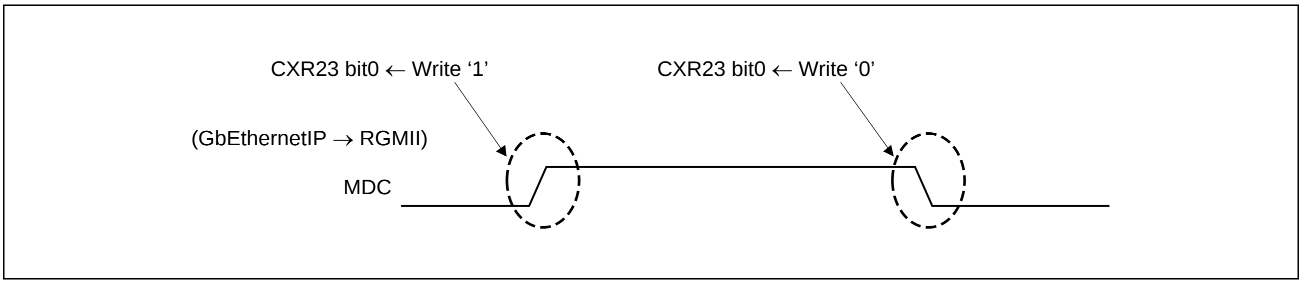


Figure 30.51 Relationship between CXR23 Write and MDC signal

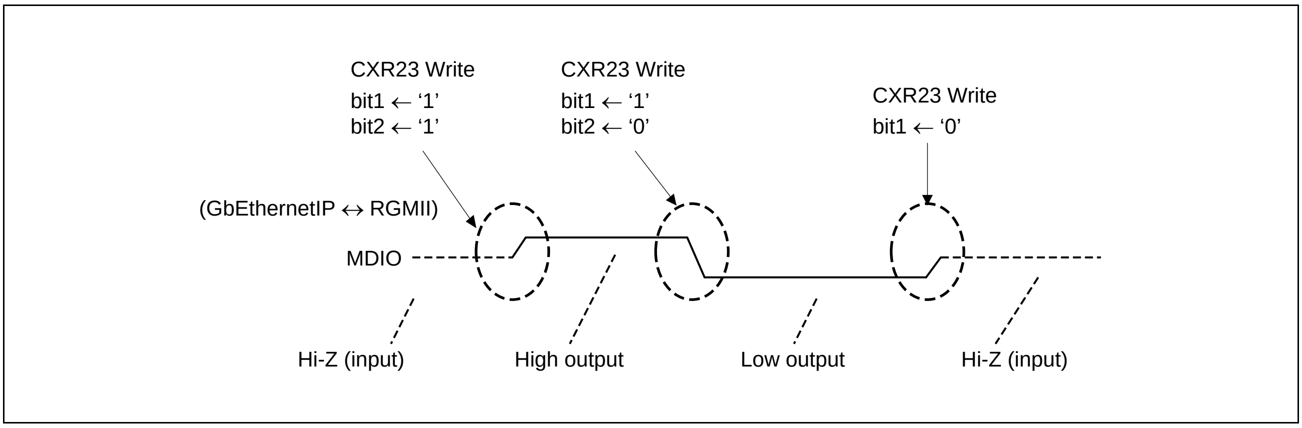


Figure 30.52 Relationship between CXR23 Write and MDIO (output) signal

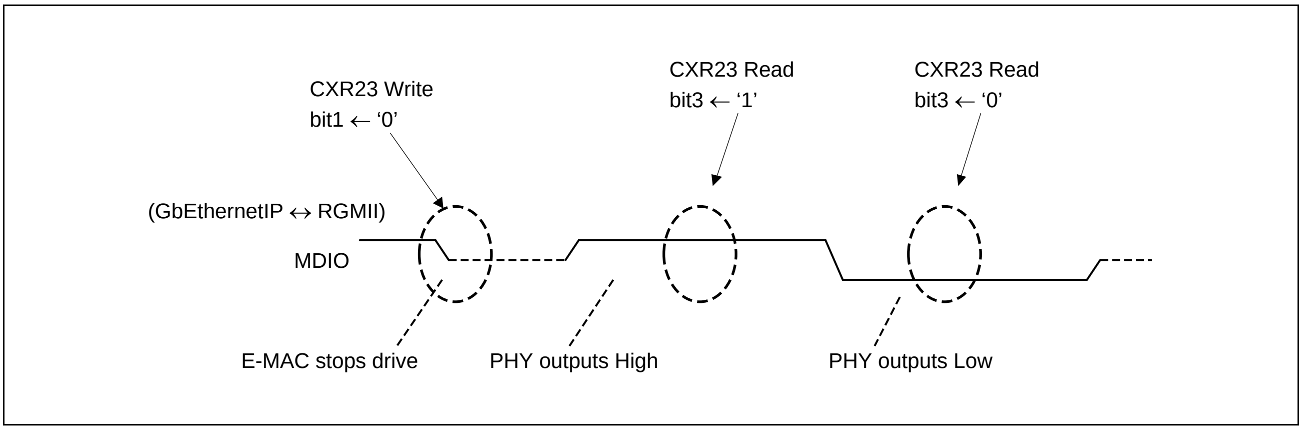


Figure 30.53 Relationship between MDIO input and CXR23 Read Values

PHY Register Write Access Procedure (Using CXR23)

To write PHY chip register, control MDC and MDIO signals based on the format given below.

(1) PHY Write Access Format

Table 30.24 PHY Write Access Format

#	Field name	I/O	Details	Remarks
1	Preamble	O	1 = 32 bits	Preamble
2	ST	O	“01”	Start
3	Op Code (Write)	O	“01”	Operation code for write
4	PHY Address	O	5 bit	PHY address (See data sheet of PHY LSI)
5	Reg Address	O	5 bit	Register address (See data sheet of PHY LSI)
6	Turn Around	O	“10”	Turnaround
7	Data (Write)	O	16 bit	Write data
8	IDLE	—	—	Idle status

(2) PHY Register Write Access Time chart

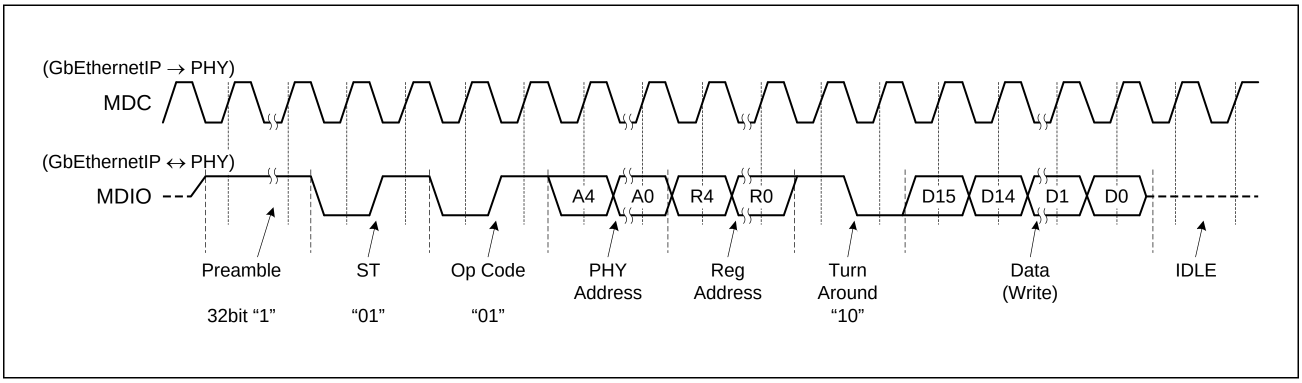


Figure 30.54 PHY Register Write Access Time Chart

(3) PHY Register Write Access Procedure

Table 30.25 PHY Register Write Access Procedure (1/2)

	CXR23					
#	bit 3	bit 2	bit 1	bit 0	CXR23 access	Description
0	0	0	0	0	Initial state	IDLE
1	0	1	1	1	CXR23 WRITE	Start of Write access (MDC rise)
2	0	1	1	0	CXR23 WRITE	(MDC fall)
3	0	1	1	1	CXR23 WRITE	Preamble1
4	0	1	1	0	CXR23 WRITE	
5	0	1	1	1	CXR23 WRITE	Preamble2
6	0	1	1	0	CXR23 WRITE	
7	0	1	1	1	CXR23 WRITE	Preamble3
8	0	1	1	0	CXR23 WRITE	
—	—	—	—	—	—	Repeat
64	0	1	1	0	CXR23 WRITE	
65	0	1	1	1	CXR23 WRITE	Preamble32
66	0	0	1	0	CXR23 WRITE	
67	0	0	1	1	CXR23 WRITE	ST
68	0	1	1	0	CXR23 WRITE	
69	0	1	1	1	CXR23 WRITE	ST
70	0	0	1	0	CXR23 WRITE	
71	0	0	1	1	CXR23 WRITE	Op Code (Write)
72	0	1	1	0	CXR23 WRITE	
73	0	1	1	1	CXR23 WRITE	Op Code (Write)
74	0	A4	1	0	CXR23 WRITE	PHY address A4 bit is output here
75	0	A4	1	1	CXR23 WRITE	PHY Address
76	0	A3	1	0	CXR23 WRITE	
77	0	A3	1	1	CXR23 WRITE	PHY Address
78	0	A2	1	0	CXR23 WRITE	
79	0	A2	1	1	CXR23 WRITE	PHY Address

Table 30.25 PHY Register Write Access Procedure (2/2)

#	CXR23				CXR23 access	Description
	bit 3	bit 2	bit 1	bit 0		
80	0	A1	1	0	CXR23 WRITE	
81	0	A1	1	1	CXR23 WRITE	PHY Address
82	0	A0	1	0	CXR23 WRITE	
83	0	A0	1	1	CXR23 WRITE	PHY Address
84	0	R4	1	0	CXR23 WRITE	REG address A4 bit is output here
85	0	R4	1	1	CXR23 WRITE	REG Address
86	0	R3	1	0	CXR23 WRITE	
87	0	R3	1	1	CXR23 WRITE	REG Address
88	0	R2	1	0	CXR23 WRITE	
89	0	R2	1	1	CXR23 WRITE	REG Address
90	0	R1	1	0	CXR23 WRITE	
91	0	R1	1	1	CXR23 WRITE	REG Address
92	0	R0	1	0	CXR23 WRITE	
93	0	R0	1	1	CXR23 WRITE	REG Address
94	0	1	1	0	CXR23 WRITE	Turn Around
95	0	1	1	1	CXR23 WRITE	Turn Around
96	0	0	1	0	CXR23 WRITE	Turn Around
97	0	0	1	1	CXR23 WRITE	Turn Around
98	0	D15	1	0	CXR23 WRITE	Write data D15 bit is output here
99	0	D15	1	1	CXR23 WRITE	Write Data
100	0	D14	1	0	CXR23 WRITE	
101	0	D14	1	1	CXR23 WRITE	Write Data
—	—	—	—	—	—	Repeat
112	0	D1	1	0	CXR23 WRITE	
113	0	D1	1	1	CXR23 WRITE	Write Data
114	0	D0	1	0	CXR23 WRITE	
115	0	D0	1	1	CXR23 WRITE	Write Data
116	0	0	0	0	CXR23 WRITE	IDLE
117	0	0	0	1	CXR23 WRITE	IDLE
118	0	0	0	0	CXR23 WRITE	IDLE
119	0	0	0	1	CXR23 WRITE	IDLE
120	0	0	0	0	CXR23 WRITE	IDLE
121	0	0	0	1	CXR23 WRITE	IDLE
122	0	0	0	0	CXR23 WRITE	IDLE End of write access

PHY Register Read Access Procedure (Using CXR23)

To read from the PHY chip registers, control MDC and MDIO signals in the format given below

(1) PHY Read Access Format

Table 30.26 PHY Read Access Format

#	Field name	I/O	Details	Remarks
1	Preamble	O	1 = 32 bits	Preamble
2	ST	O	“01”	Start
3	Op Code (read)	O	“10”	Operation code for read
4	PHY Address	O	5 bit	PHY address (See data sheet of PHY LSI)
5	Reg Address	O	5 bit	Register address (See data sheet of PHY LSI)
6	Turn Around	I	“Z0”	Turnaround (input/output switching)
7	Data (Read)	I	16 bit	Read data
8	IDLE	—	—	Idle state

(2) PHY Register Read Access Time Chart

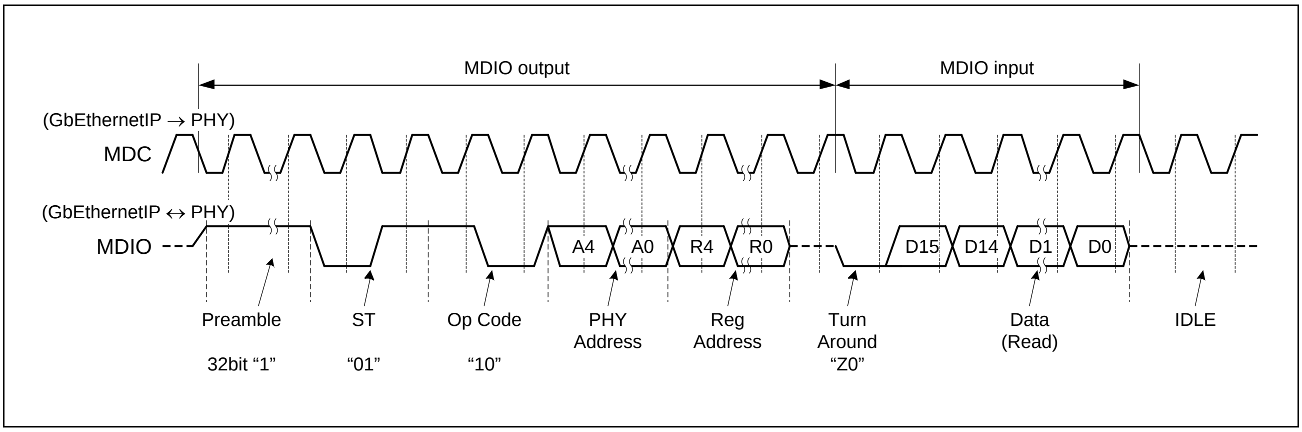


Figure 30.55 PHY Register Read Access Time Chart

(3) PHY Register Read Access Procedure

Table 30.27 PHY Register Read Access Procedure (1/2)

#	CXR23				CXR23 access	Description
	bit 3	bit 2	bit 1	bit 0		
0	0	0	0	0	Initial state	IDLE
1	0	1	1	1	CXR23 WRITE	Start of write access (MDC rise)
2	0	1	1	0	CXR23 WRITE	(MDC fall)
3	0	1	1	1	CXR23 WRITE	Preamble1
4	0	1	1	0	CXR23 WRITE	
5	0	1	1	1	CXR23 WRITE	Preamble2
6	0	1	1	0	CXR23 WRITE	
7	0	1	1	1	CXR23 WRITE	Preamble3
8	0	1	1	0	CXR23 WRITE	
—	—	—	—	—	—	Repeat
64	0	1	1	0	CXR23 WRITE	
65	0	1	1	1	CXR23 WRITE	Preamble32
66	0	0	1	0	CXR23 WRITE	
67	0	0	1	1	CXR23 WRITE	ST
68	0	1	1	0	CXR23 WRITE	
69	0	1	1	1	CXR23 WRITE	ST
70	0	1	1	0	CXR23 WRITE	
71	0	1	1	1	CXR23 WRITE	Op Code (Read)
72	0	0	1	0	CXR23 WRITE	
73	0	0	1	1	CXR23 WRITE	Op Code (Read)
74	0	A4	1	0	CXR23 WRITE	PHY address A4 bit is output here
75	0	A4	1	1	CXR23 WRITE	PHY Address
76	0	A3	1	0	CXR23 WRITE	
77	0	A3	1	1	CXR23 WRITE	PHY Address
78	0	A2	1	0	CXR23 WRITE	
79	0	A2	1	1	CXR23 WRITE	PHY Address
80	0	A1	1	0	CXR23 WRITE	
81	0	A1	1	1	CXR23 WRITE	PHY Address
82	0	A0	1	0	CXR23 WRITE	
83	0	A0	1	1	CXR23 WRITE	PHY Address
84	0	R4	1	0	CXR23 WRITE	REG address A4 bit is output here
85	0	R4	1	1	CXR23 WRITE	REG Address
86	0	R3	1	0	CXR23 WRITE	
87	0	R3	1	1	CXR23 WRITE	REG Address
88	0	R2	1	0	CXR23 WRITE	
89	0	R2	1	1	CXR23 WRITE	REG Address
90	0	R1	1	0	CXR23 WRITE	
91	0	R1	1	1	CXR23 WRITE	REG Address
92	0	R0	1	0	CXR23 WRITE	
93	0	R0	1	1	CXR23 WRITE	REG Address
94	0	0	0	0	CXR23 WRITE	Turn Around (MDIO Hi-Z Output MDIO Output -> Input)

Table 30.27 PHY Register Read Access Procedure (2/2)

#	CXR23				CXR23 access	Description
	bit 3	bit 2	bit 1	bit 0		
95	0	0	0	1	CXR23 WRITE	Turn Around
96	0	0	0	0	CXR23 WRITE	Turn Around
97	0	0	0	1	CXR23 WRITE	Turn Around
98	0	0	0	0	CXR23 WRITE	Read Data Start
99	D15	X	X	X	CXR23 Read	Here Data bit15 in PHY register can be seen in bit3 in CXR23
100	0	0	0	1	CXR23 WRITE	
101	0	0	0	0	CXR23 WRITE	
102	D14	X	X	X	CXR23 Read	Data bit14 read
103	0	0	0	1	CXR23 WRITE	
104	0	0	0	0	CXR23 WRITE	
—	—	—	—	—	—	Repeat
141	D1	X	X	X	CXR23 Read	Data bit1 read
142	0	0	0	1	CXR23 WRITE	
143	0	0	0	0	CXR23 WRITE	
144	D1	X	X	X	CXR23 Read	Data bit0 read
145	0	0	0	1	CXR23 WRITE	Read Data End
146	0	0	0	0	CXR23 WRITE	IDLE
147	0	0	0	1	CXR23 WRITE	IDLE
148	0	0	0	0	CXR23 WRITE	IDLE
149	0	0	0	1	CXR23 WRITE	IDLE
150	0	0	0	0	CXR23 WRITE	IDLE
151	0	0	0	1	CXR23 WRITE	IDLE
152	0	0	0	0	CXR23 WRITE	IDLE End of read access

31. A/D Converter

31.1 Features

This LSI has a successive approximation A/D converter with a 12-bit accuracy. Up to eight analog input channels can be selected.

■ Resolution

12 bits

■ Eight input channels

■ Input voltage range

0 V to 1.8 V

■ Minimum conversion time

1.0 µs per channel (when A/D conversion clock ADIVCLK is 20 MHz)

■ Eight data registers

The A/D conversion result is stored in a 32-bit data register corresponding to each channel.

■ Sample-and-hold function

■ Trigger mode

- Software trigger mode and hardware trigger mode are available.
- Software trigger mode to start A/D conversion by software
- Hardware trigger mode to start A/D conversion by an asynchronous trigger or synchronous trigger
  - Asynchronous trigger using an external pin (ADC_TRG) as the activation source
  - Synchronous trigger using the multi-function timer pulse unit 3 (MTU3a) or general-purpose
  - PWM timer (GPT) as the activation source

■ Operating mode

- Select mode and scan mode are available.
- Select mode to convert the specified single analog input channel
  - 1-buffer mode or 4-buffer mode can be specified for storing the A/D conversion result.
  - In 4-buffer mode, A/D conversion proceeds four times for the selected analog input and the A/D conversion results are stored in four registers.
- Scan mode to convert the analog inputs of arbitrarily selected channels in ascending order of channel number
  - 1-buffer mode can be specified for storing the A/D conversion result.

■ Conversion mode

- Single mode and repeat mode are available.
- Single mode to proceed A/D conversion only once
- Repeat mode to repeatedly proceed A/D conversion

■ A single interrupt source

- An A/D conversion end interrupt (INTAD) can be generated on completion of A/D conversion.
  - In the select mode with 1-buffer mode selected, an A/D conversion end interrupt is generated on completion of a single conversion.
  - In the select mode with 4-buffer mode selected, an A/D conversion end interrupt is generated on completion of four rounds of conversion.
  - In scan mode, an A/D conversion end interrupt is generated on completion of scanning of all the selected channels.

Table 31.1 shows the outline of the functions of the A/D converter.

Table 31.1 Functions of A/D Converter (1/2)

Item				Source
Analog Input Channel				ADC_CH0 to ADC_CH7
Conditions for starting A/D conversion	Software	Software trigger		(Enabled by software)
	Asynchronous trigger	External trigger*1	Trigger input pin	ADC_TRG
	Synchronous trigger	Trigger from MTU3a	Compare match with or input capture to MTU0.TGRA	TRGA0N
			Compare match with or input capture to MTU1.TGRA	TRGA1N
			Compare match with or input capture to MTU2.TGRA	TRGA2N
			Compare match with or input capture to MTU3.TGRA	TRGA3N
			Compare match with or input capture to MTU4.TGRA or underflow of MTU4.TCNT in complementary PWM mode	TRGA4N
			Compare match with or input capture to MTU6.TGRA	TRGA6N
			Compare match with or input capture to MTU7.TGRA or underflow of MTU7.TCNT in complementary PWM mode	TRGA7N
			Compare match with MTU0.TGRE	TRG0N
			Compare match between MTU4.TADCORA and MTU4.TCNT	TRG4AN
			Compare match between MTU4.TADCORB and MTU4.TCNT	TRG4BN
			Compare match between MTU4.TADCORA and MTU4.TCNT or compare match between MTU4.TADCORB and MTU4.TCNT	TRG4AN or TRG4BN
			Compare match between MTU4.TADCORA and MTU4.TCNT and compare match between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is used)	TRG4ABN
			Compare match between MTU7.TADCORA and MTU7.TCNT	TRG7AN
			Compare match between MTU7.TADCORB and MTU7.TCNT	TRG7BN
			Compare match between MTU7.TADCORA and MTU7.TCNT or compare match between MTU7.TADCORB and MTU7.TCNT compare match	TRG7AN or TRG7BN
			Compare match between MTU7.TADCORA and MTU7.TCNT and compare match between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is used)	TRG7ABN

Table 31.1 Functions of A/D Converter (2/2)

Item				Source
Conditions for starting A/D conversion	Synchronous trigger	Trigger from GPT	Compare match with GPT0.GTADTRA	ADTRGA0
			Compare match with GPT0.GTADTRB	ADTRGB0
			Compare match with GPT1.GTADTRA	ADTRGA1
			Compare match with GPT1.GTADTRB	ADTRGB1
			Compare match with GPT2.GTADTRA	ADTRGA2
			Compare match with GPT2.GTADTRB	ADTRGB2
			Compare match with GPT3.GTADTRA	ADTRGA3
			Compare match with GPT3.GTADTRB	ADTRGB3
			Compare match with GPT0.GTADTRA or GPT0.GTADTRB	ADTRGA0 or ADTRGB0
			Compare match with GPT1.GTADTRA or GPT1.GTADTRB	ADTRGA1 or ADTRGB1
			Compare match with GPT2.GTADTRA or GPT2.GTADTRB	ADTRGA2 or DTRGB2
			Compare match with GPT3.GTADTRA or GPT3.GTADTRB	ADTRGA3 or ADTRGB3
Interrupt				INTAD

Note 1. To set ADC_TRG as the trigger for starting A/D conversion, set the general-purpose I/O function. For details, see **Section 41, General Purpose Input Output Port**.

Table 31.2 lists the I/O pins used in the A/D converter.

Table 31.2 I/O Pins of A/D Converter

Pin Name	I/O	Name	Function
ADC_CH0 to ADC_CH7	Input	Analog input pins	Analog input pins
ADC_TRG	Input	A/D conversion trigger input	External trigger input pin for starting A/D conversion
ADC_AVDD18	Input	Analog power supply voltage	A/D converter power supply pin
VSS	Input	Analog ground	A/D converter ground pin

31.2 List of Registers

Table 31.3 shows the register list of this module.

The base address is as follows.

Base address: H’0_1005_9000 (Cortex-A55 Address Space)
Base address: H’4005_9000 (Cortex-M33 Address Space Non-Secure)
Base address: H’5005_9000 (Cortex-M33 Address Space Secure)

Table 31.3 List of Registers of A/D Converter

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
A/D converter mode register 0	ADM0	R/W	H’0000	H’0	32
A/D converter mode register 1	ADM1	R/W	H’0000	H’4	32
A/D converter mode register 2	ADM2	R/W	H’0000	H’8	32
A/D converter mode register 3	ADM3	R/W	H’0000	H’C	32
Reserved	—	R	H’0000	H’10 – 1F	32
A/D converter interrupt control register	ADINT	R/W	H’0000	H’20	32
A/D converter status register	ADSTS	R/W	H’0000	H’24	32
A/D converter clock division setting register	ADIVC	R/W	H’0000	H’28	32
A/D converter external trigger pin filter control register	ADFIL	R/W	H’0000	H’2C	32
A/D conversion result register 0	ADCR0	R	H’0000	H’30	32
A/D conversion result register 1	ADCR1	R	H’0000	H’34	32
A/D conversion result register 2	ADCR2	R	H’0000	H’38	32
A/D conversion result register 3	ADCR3	R	H’0000	H’3C	32
A/D conversion result register 4	ADCR4	R	H’0000	H’40	32
A/D conversion result register 5	ADCR5	R	H’0000	H’44	32
A/D conversion result register 6	ADCR6	R	H’0000	H’48	32
A/D conversion result register 7	ADCR7	R	H’0000	H’4C	32
Reserved	—	R	H’0000	H’50 – 6C	32
Renesas reserved area*[1]	—	R/W	Undefined	H’70 – 7F	32

Note: When writing successively to the same register, the notes in **Section 31.5.2, Timing Restrictions**, need to be observed.

Note 1. Writing to a Renesas reserved area is prohibited.

31.3 Register Descriptions

31.3.1 A/D Converter Mode Register 0 (ADM0)

The ADM0 is a 32-bit register that controls A/D conversion and the power-saving mode of the A/D converter, and also executes a software reset. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRESB	—	—	—	—	—	—	—	—	—	—	—	—	PWDWNB	ADBSY	ADCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
15	SRESB	0	R/W	Issues a software reset to the A/D converter. Circuits (e.g. registers) other than those of the A/D converter will not be reset. To release the A/D converter from the reset state, write 1 to this bit. 0: A/D converter is reset. 1: A/D converter is released from the reset state. The A/D converter enters the power-saving mode when this bit is used to execute a reset.
14 to 3	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
2	PWDWNB	0	R/W	Specifies the A/D converter to enter or exit the power-saving mode. 0: Power-saving mode 1: Normal mode
1	ADBSY	0	R	Indicates the status of the A/D converter. This bit is read-only. Writing to this bit is ignored. 0: A/D converter is stopped. 1: A/D converter is busy. For operation of the ADBSY bit, see **Figure 31.1**.
0	ADCE	0	R/W	Stops or enables the A/D converter. 0: A/D converter is stopped. 1: A/D converter is enabled. When 0 is written to the ADCE bit, the A/D converter stops operating. For operation of the ADCE bit at reading, see **Figure 31.1**.

Note: Notes regarding the ADM0 register are given on the next page.

NOTES

1. To enable A/D conversion operation using the ADCE bit, first switch from PWDWNB = 0 (power-saving mode) to PWDWNB = 1 (normal mode) and then wait for a period equal to or longer than the stabilization wait time before setting the ADCE bit to 1. The stabilization wait time is 1 µs.
2. To make a transition to the power-saving mode using the PWDWNB bit, first stop operation of the A/D converter (ADCE = 0) and then confirm that A/D conversion is completed (ADBSY = 0) before setting the PWDWNB bit to 0.

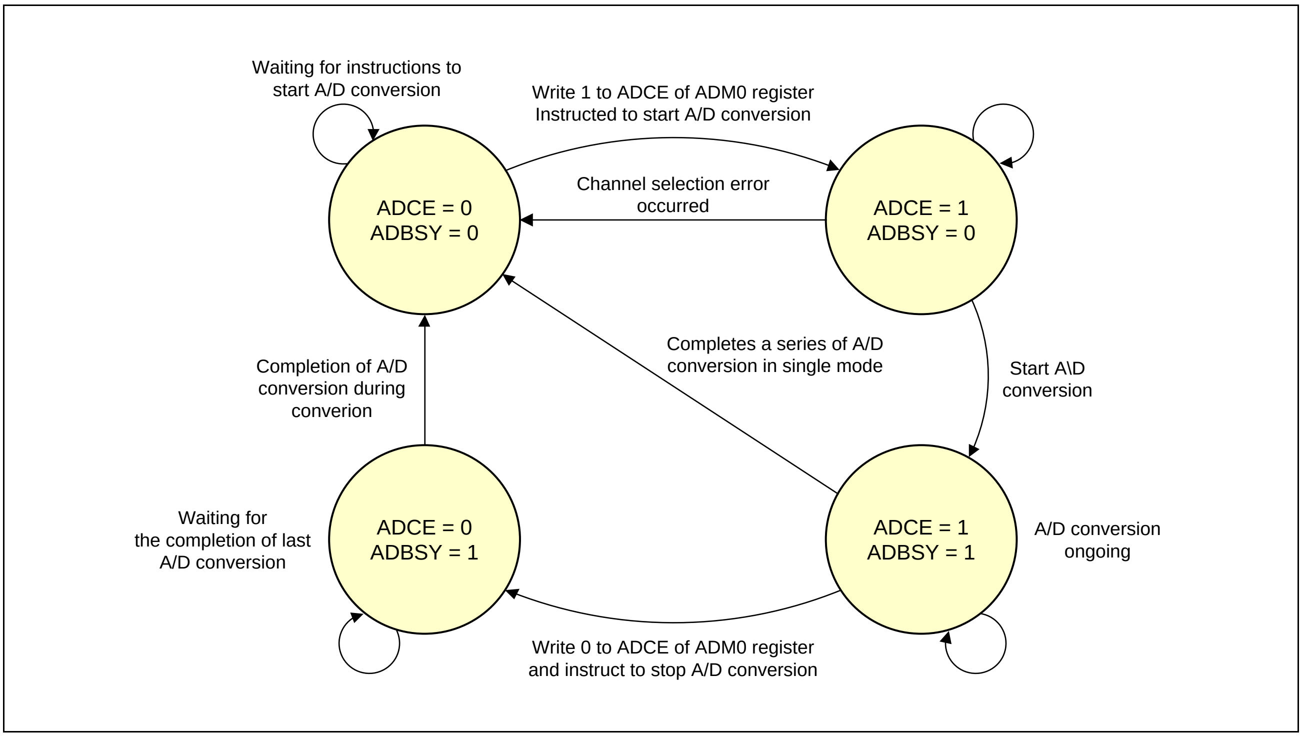


Figure 31.1 State Transitions of ADCE Bit and ADBSY Bit

31.3.2 A/D Converter Mode Register 1 (ADM1)

The ADM1 is a 32-bit register that controls A/D conversion and sets the mode for a hardware trigger. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	TRGEN 5	TRGEN 4	TRGEN 3	TRGEN 2	TRGEN 1	TRGEN 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	EGA1	EGA0	—	—	—	—	—	—	—	BS	RPS	MS	TRGIN	TRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31 to 22</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
When read, the initial value is read. The written value will be ignored.</td></tr>
<tr><td>21 to 16</td><td>TRGEN[5:0]</td><td>All 0</td><td>R/W</td><td>These bits select the hardware trigger pin.
For details, see Table 31.4.</td></tr>
<tr><td>15, 14</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
When read, the initial value is read. The written value will be ignored.</td></tr>
<tr><td>13, 12</td><td>EGA[1:0]</td><td>00</td><td>R/W</td><td>These bits select the valid edge of the trigger pin.
<table>
<tr><th>EGA1</th><th>EGA0</th><th>Valid Edge</th></tr>
<tr><td>0</td><td>0</td><td>Hardware trigger is invalid</td></tr>
<tr><td>0</td><td>1</td><td>Falling edge</td></tr>
<tr><td>1</td><td>0</td><td>Rising edge</td></tr>
<tr><td>1</td><td>1</td><td>Both edges</td></tr>
</table>
<i>Note</i>: When the EGA[1:0] bits are set to 00 in hardware trigger mode, a trigger from the trigger pin will be ignored. When a trigger pin of the MTU3a or GPT is used, the falling edge is always valid and the setting of the EGA[1:0] bits (01b, 10b, or 11b) will be ignored. When the external trigger pin (ADC_TRG) is used, the valid edge is set by the EGA[1:0] bits.</td></tr>
<tr><td>11 to 5</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
When read, the initial value is read. The written value will be ignored.</td></tr>
<tr><td>4</td><td>BS</td><td>0</td><td>R/W</td><td>Selects the buffer mode (valid in select mode).
0: 1-buffer mode
1: 4-buffer mode
4-buffer mode is prohibited in scan mode.</td></tr>
<tr><td>3</td><td>RPS</td><td>0</td><td>R/W</td><td>Selects the repeat number.
0: Single
1: Repeat</td></tr>
<tr><td>2</td><td>MS</td><td>0</td><td>R/W</td><td>Selects the operating mode.
0: Scan mode
1: Select mode</td></tr>
<tr><td>1</td><td>TRGIN</td><td>0</td><td>R/W</td><td>Selects the trigger input mode.
0: Auto mode
1: Step mode</td></tr>
</table>

Bit	Bit Name	Initial Value	R/W	Description
0	TRG	0	R/W	Sets the trigger mode. 0: Software trigger mode*[1] 1: Hardware trigger mode

Note 1. Set the EGA[1:0] bits to 00 when using software trigger mode.

Table 31.4 TRGEN[5:0] Bits and Trigger Source

Module	Source	Description	TRGEN[5:0] Bits					
			5	4	3	2	1	0
External pin	ADC_TRG	External trigger input	0	0	0	0	0	0
MTU	TRGA0N	Compare match with or input capture to MTU0.TGRA	0	0	0	0	0	1
	TRGA1N	Compare match with or input capture to MTU1.TGRA	0	0	0	0	1	0
	TRGA2N	Compare match with or input capture to MTU2.TGRA	0	0	0	0	1	1
	TRGA3N	Compare match with or input capture to MTU3.TGRA	0	0	0	1	0	0
	TRGA4N	Compare match with or input capture to MTU4.TGRA or underflow (trough) of MTU4.TCNT in complementary PWM mode	0	0	0	1	0	1
	TRGA6N	Compare match with or input capture to MTU6.TGRA	0	0	0	1	1	0
	TRGA7N	Compare match with or input capture to MTU7.TGRA or underflow (trough) of MTU7.TCNT in complementary PWM mode	0	0	0	1	1	1
	TRG0N	Compare match with MTU0.TGRE	0	0	1	0	0	0
	TRG4AN	Compare match between MTU4.TADCORA and MTU4.TCNT	0	0	1	0	0	1
	TRG4BN	Compare match between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	0
	TRG4AN or TRG4BN	Compare match between MTU4.TADCORA and MTU4.TCNT or compare match between MTU4.TADCORB and MTU4.TCNT	0	0	1	0	1	1
	TRG4ABN	Compare match between MTU4.TADCORA and MTU4.TCNT and compare match between MTU4.TADCORB and MTU4.TCNT (when interrupt skipping function 2 is used)	0	0	1	1	0	0
	TRG7AN	Compare match between MTU7.TADCORA and MTU7.TCNT	0	0	1	1	0	1
	TRG7BN	Compare match between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	0
	TRG7AN or TRG7BN	Compare match between MTU7.TADCORA and MTU7.TCNT or compare match between MTU7.TADCORB and MTU7.TCNT	0	0	1	1	1	1
	TRG7ABN	Compare match between MTU7.TADCORA and MTU7.TCNT and compare match between MTU7.TADCORB and MTU7.TCNT (when interrupt skipping function 2 is used)	0	1	0	0	0	0
GPT	ADTRGA0	Compare match with GPT0.GTADTRA	0	1	0	0	0	1
	ADTRGB0	Compare match with GPT0.GTADTRB	0	1	0	0	1	0
	ADTRGA1	Compare match with GPT1.GTADTRA	0	1	0	0	1	1
	ADTRGB1	Compare match with GPT1.GTADTRB	0	1	0	1	0	0
	ADTRGA2	Compare match with GPT2.GTADTRA	0	1	0	1	0	1
	ADTRGB2	Compare match with GPT2.GTADTRB	0	1	0	1	1	0
	ADTRGA3	Compare match with GPT3.GTADTRA	0	1	0	1	1	1
	ADTRGB3	Compare match with GPT3.GTADTRB	0	1	1	0	0	0
	ADTRGA0 or ADTRGB0	Compare match with GPT0.GTADTRA or GPT0.GTADTRB	0	1	1	0	0	1
	ADTRGA1 or ADTRGB1	Compare match with GPT1.GTADTRA or GPT1.GTADTRB	0	1	1	0	1	0
	ADTRGA2 or ADTRGB2	Compare match with GPT2.GTADTRA or GPT2.GTADTRB	0	1	1	0	1	1
	ADTRGA3 or ADTRGB3	Compare match with GPT3.GTADTRA or GPT3.GTADTRB	0	1	1	1	0	0

31.3.3 A/D Converter Mode Register 2 (ADM2)

The ADM2 is a 32-bit register that specifies the analog input channels to be used in A/D conversion. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CHSEL 7	CHSEL 6	CHSEL 5	CHSEL 4	CHSEL 3	CHSEL 2	CHSEL 1	CHSEL 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 0	CHSEL[7:0]	All 0	R/W	These bits select the A/D analog input channel n (n = 0 to 7). 0: Analog input channel n is not subjected to conversion. 1: Analog input channel n is subjected to conversion.

Note 1. Select only one channel in select mode.

Note 2. If more than one channel is selected in select mode, a channel select error occurs.

31.3.4 A/D Converter Mode Register 3 (ADM3)

The ADM3 is a 32-bit register that sets the sampling period of the A/D converter. Be sure to set this register before starting the A/D converter. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADIL7	ADIL6	ADIL5	ADIL4	ADIL3	ADIL2	ADIL1	ADIL0	ADCMP 7	ADCMP 6	ADCMP 5	ADCMP 4	ADCMP 3	ADCMP 2	ADCMP 1	ADCMP 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADSMP 15	ADSMP 14	ADSMP 13	ADSMP 12	ADSMP 11	ADSMP 10	ADSMP 9	ADSMP 8	ADSMP 7	ADSMP 6	ADSMP 5	ADSMP 4	ADSMP 3	ADSMP 2	ADSMP 1	ADSMP 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADIL[7:0]	All 0	R/W	These bits should be set to H'00.
23 to 16	ADCMP[7:0]	All 0	R/W	These bits should be set to H'0E.
15 to 0	ADSMP[15:0]	All 0	R/W	These bits set the sampling period (unit: ADIVCLK cycle). A value from H'06 (6 T) to H'AF0 (2800 T) can be set. T stands for the cycle of ADIVCLK (20 MHz).

Note: The ADIL[7:0] bits should be set to H'00 and the ADCMP[7:0] bits should be set to H'0E. If they are set to other values, normal operation cannot be guaranteed.

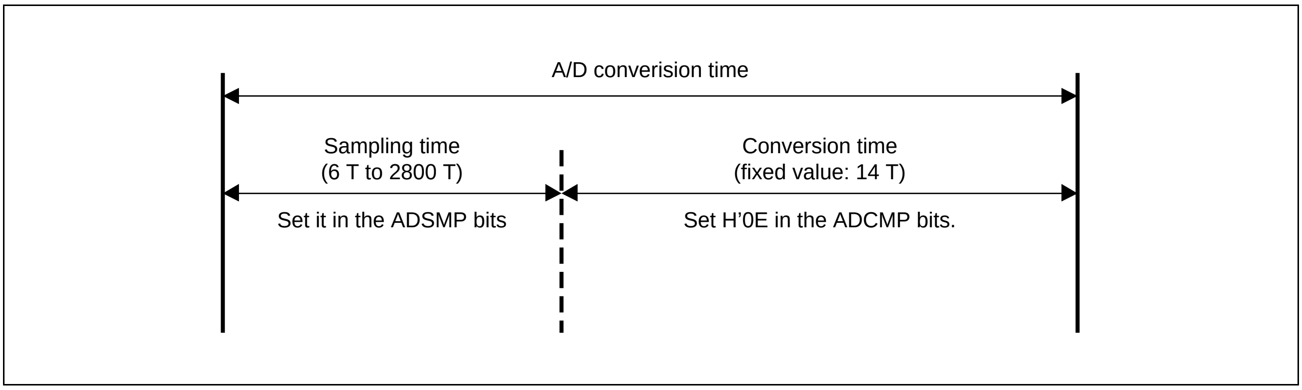


Figure 31.2 Correspondence between ADM3 Register and A/D Conversion Time

NOTE

ADIVCLK is a clock obtained by dividing the frequency of ADC_ADCLK(TSUϕ) with the ADIVC register setting.

31.3.5 A/D Converter Interrupt Control Register (ADINT)

The ADINT is a 32-bit register that controls interrupts. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INTS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CSEEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	INTEN7	INTEN6	INTEN5	INTEN4	INTEN3	INTEN2	INTEN1	INTEN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	INTS	0	R/W	Selects the attribute of the interrupt (INTAD) signal. 0: Pulse signal 1: Reserved This bit should be set to 0.
30 to 17	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
16	CSEEN	0	R/W	Enables or disables the A/D conversion channel select error interrupt. 0: Interrupt output is disabled. 1: Interrupt output is enabled.
15 to 8	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 0	INTEN[7:0]	All 0	R/W	These bits enable or disable the conversion end interrupt of channel n (n = 0 to 7). 0: Interrupt output is disabled. 1: Interrupt output is enabled.

The interrupt signal is specified as a pulse and that pulse is output at any of the following timing:

1. An A/D conversion request is detected while the ADM2 register has caused an A/D conversion channel select error
2. Completion of conversion on the channel for which interrupt output has been enabled in the ADINT register
3. An A/D conversion request is detected while 4-buffer mode has been specified in scan mode

31.3.6 A/D Converter Status Register (ADSTS)

The ADSTS is a 32-bit register that controls the state of the A/D converter. This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TRGS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CSEST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	INTST7	INTST6	INTST5	INTST4	INTST3	INTST2	INTST1	INTST0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TRGS	0	R/W	• When reading 0: Nothing detected 1: A trigger was detected during the A/D conversion time in **Figure 31.2**. • When writing 0: No effect 1: The state is cleared.
30 to 17	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
16	CSEST	0	R/W	Indicates the status of the A/D conversion channel select error interrupt. It is the status of the interrupt for an error that occurs when A/D conversion is started with no channel being specified in the CHSEL[7:0] bits in the ADM2 register as a channel on which A/D conversion proceeds. • When reading 0: No A/D conversion channel select error occurred. 1: An A/D conversion channel select error occurred. • When writing 0: No effect 1: The status is cleared.
15 to 8	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
7 to 0	INTST[7:0]	All 0	R/W	Indicates the state of the conversion end interrupt of channel n (n = 0 to 7). • When reading 0: Conversion has not finished. 1: Conversion has finished. • When writing 0: No effect 1: The state is cleared.*1

Note 1. When an interrupt source is generated simultaneously with clearing of the interrupt source, clearing is ignored and the respective bit remains set to 1.

31.3.7 A/D Converter Clock Division Setting Register (ADIVC)

The ADIVC is a 32-bit register that sets the frequency division ratio for dividing the frequency of ADC_ADCLK(TSUϕ) and supplying a clock to the A/D converter. ADIVCLK is a clock obtained by dividing the frequency of ADC_ADCLK(TSUϕ).

This register can be read from and written to in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DIVADC 8	DIVADC 7	DIVADC 6	DIVADC 5	DIVADC 4	DIVADC 3	DIVADC 2	DIVADC 1	DIVADC 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
8 to 0	DIVADC[8:0]	All 0	R/W	These bits set the frequency division ratio. Only division by 4 can be set, and any other setting is prohibited. 000000100: Division by 4

31.3.8 A/D Converter External Trigger Pin Filter Control Register (ADFIL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	FILNUM		—	—	—	FILONOFF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
5, 4	FILNUM[1:0]	00	R/W	These bits select the number of stages of the AD external trigger pin filter. The signal to be filtered is ADC_ADCLK(TSUϕ) (80 MHz). 00: 12.5 ns (80 MHz) × 4 stages 01: 12.5 ns (80 MHz) × 8 stages 10: 12.5 ns (80 MHz) × 12 stages 11: 12.5 ns (80 MHz) × 16 stages
3 to 1	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
0	FILONOFF	0	R/W	Enables or disables the AD external trigger pin filter 0: Filter is disabled. 1: Filter is enabled.

31.3.9 A/D Conversion Result Registers 7 to 0 (ADCR7 to ADCR0)

The ADCR7 to ADCR0 are 32-bit registers that hold the A/D conversion results. The A/D converter has eight 32-bit registers. These registers are read from in 32-bit units.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
11 to 0	AD[11:0]	All 0	R	These bits hold the A/D conversion results.

Analog Input Channel	ADCR7 to ADCR0 Registers	
	Select Mode with 1-Buffer Mode Selected or Scan Mode	Select Mode with 4-Buffer Mode Selected
0	ADCR0	ADCR3 to ADCR0
1	ADCR1	
2	ADCR2	
3	ADCR3	
4	ADCR4	ADCR7 to ADCR4
5	ADCR5	
6	ADCR6	
7	ADCR7	

The following formula indicates the relationship between the analog input voltage that is input to analog input pins (ADC_CH7 to ADC_CH0) and the A/D conversion results (A/D conversion result registers (ADCR7 to ADCR0)).

$$\mathrm{ADCR} = \mathrm{INT}\left[\frac{\mathrm{Vin}}{\mathrm{ADC\_AVREF}} \times 2^{\mathrm{d}} \times 0.5\right]$$

or

$$(\mathrm{ADCR} - 0.5) \times \frac{\mathrm{ADC\_AVREF}}{2^{\mathrm{d}}} \leq \mathrm{Vin} < (\mathrm{ADCR} + 0.5) \times \frac{\mathrm{ADC\_AVREF}}{2^{\mathrm{d}}}$$

INT[ ]:	Function that returns the integer portion of the value enclosed in [ ]
Vin:	Analog input voltage
ADC_AVREF:	Voltage of power supply pin(ADC_AVDD18) for the analog unit
ADCR:	Value of A/D conversion result registers (ADCR7 to ADCR0)
d:	Resolution of A/D converter (d = 12 in this LSI)

The relationship between the analog input voltage and A/D conversion result is shown in the figure below.

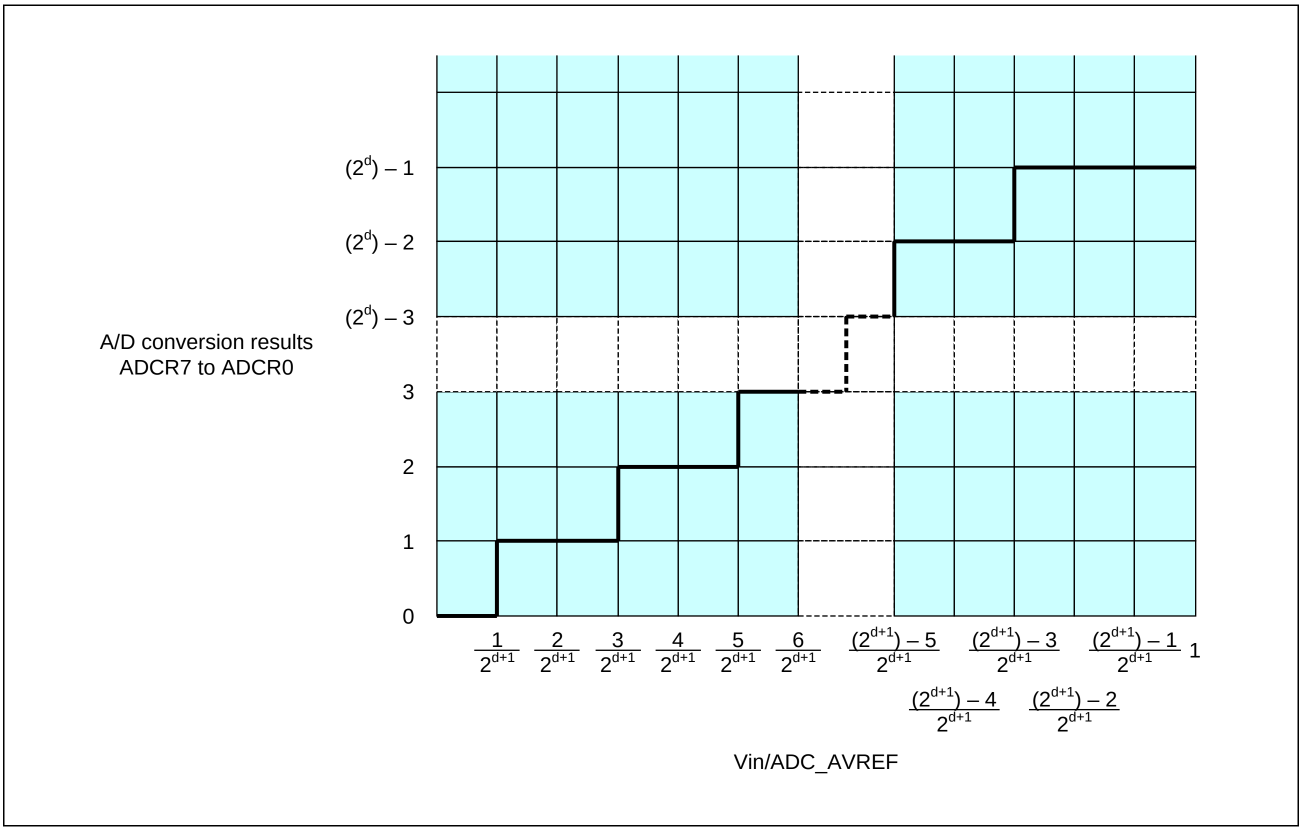


Figure 31.3 Relationship between Analog Input Voltage and A/D Conversion Result

31.4 Operation

31.4.1 Types of A/D Conversion Modes

The following A/D conversion modes can be specified. A/D conversion modes can be set in the ADM1 register.

Table 31.5 A/D Conversion Modes

Trigger Mode	Trigger Input Mode	Operating Mode	Conversion Count	Buffer Count	Operation	Example of A/D Conversion for Reference
Software trigger	—	Select	Single	1 buffer	Selected 1 channel × 1-time conversion	**Section 31.4.4.1, Example of A/D Conversion in Select Mode with Single Mode Selected**
				4 buffers	Selected 1 channel × 4-time conversion	—
			Repeat	1 buffer	(Selected 1 channel × 1-time conversion) × repeated	**Section 31.4.4.2, Example of A/D Conversion in Select Mode with Repeat Mode Selected**
				4 buffers	(Selected 1 channel × 4-time conversion) × repeated	**Section 31.4.4.3, Example of A/D Conversion in 4-Buffer Mode**
		Scan	Single	1 buffer	All selected channels × 1-time conversion	**Section 31.4.4.4, Example of A/D Conversion in Scan Mode with Single Mode Selected**
				4 buffers	(Setting prohibited)	
			Repeat	1 buffer	(All selected channels × 1-time conversion) × repeated	**Section 31.4.4.5, Example of A/D Conversion in Scan Mode with Repeat Mode Selected**
				4 buffers	(Setting prohibited)	
Hardware trigger	Auto (started with a single trigger input)	Select	Single	1 buffer	Selected 1 channel × 1-time conversion	—
				4 buffers	Selected 1 channel × 4-time conversion	—
			Repeat	1 buffer	(Selected 1 channel × 1-time conversion) × repeated	—
				4 buffers	(Selected 1 channel × 4-time conversion) × repeated	—
		Scan	Single	1 buffer	All selected channels × 1-time conversion	**Section 31.4.4.6, Example of A/D Conversion in Auto Mode**
				4 buffers	(Setting prohibited)	
			Repeat	1 buffer	(All selected channels × 1-time conversion) × repeated	—
				4 buffers	(Setting prohibited)	
	Step (conversion proceeds at every trigger input)	Select	Single	1 buffer	Selected 1 channel × 1-time conversion	—
				4 buffers	Selected 1 channel × 4-time conversion	—
			Repeat	1 buffer	(Selected 1 channel × 1-time conversion) × repeated	—
				4 buffers	(Selected 1 channel × 4-time conversion) × repeated	—
		Scan	Single	1 buffer	All selected channels × 1-time conversion	**Section 31.4.4.7, Example of A/D Conversion in Step Mode**
				4 buffers	(Setting prohibited)	
			Repeat	1 buffer	(All selected channels × 1-time conversion) × repeated	—
				4 buffers	(Setting prohibited)	

Note: When 4-buffer mode is set in scan mode, a channel select error occurs.

31.4.1.1 Trigger Modes

Software trigger mode and hardware trigger mode are the two types of trigger modes in which the trigger is the timing for starting the A/D conversion processing. These trigger modes can be set by the TRG bit in the ADM1 register.

(1) Software Trigger Mode

In software trigger mode, A/D conversion is started for the input of the ADC_CH7 to ADC_CH0 pins by setting the ADCE bit in the ADM0 register to 1. The ADBSY bit in the ADM0 register is 1 while the converter is operating.

The hardware trigger is invalid when software trigger mode is set. Set the EGA[1:0] bits in the ADM1 register to 00 when using software trigger mode. When a hardware trigger is input, A/D conversion is not started and the TRGS bit in the ADSTS register retains 0.

(2) Hardware Trigger Mode

There are two triggers in hardware trigger mode: asynchronous trigger using the ADC_TRG pin as the activation source and synchronous trigger using the multi-function timer pulse unit 3 (MTU3a) or general purpose PWM timer (GPT) as the activation source.

The valid edge of the selected hardware trigger can be set by the EGA[1:0] bits in the ADM1 register.

When the ADCE bit in the ADM0 register is set to 1, the A/D converter enters a standby state for a hardware trigger after 250 ns as passed and will start conversion operation upon detection of a valid edge. The ADBSY bit in the ADM0 register remains to be 1 during conversion operation.

The status of the TRGS bit in the ADSTS register can be cleared by writing 1 to the TRGS bit.

In hardware trigger mode, the available trigger input modes are auto mode and step mode. The trigger input mode can be set in the TRGIN bit in the ADM1 register.

(a) Auto mode

When a hardware trigger is input once, A/D conversion is repeated automatically for the conversion count in accordance with the specified conversion settings.

(b) Step mode

A/D conversion proceeds every time a hardware trigger is input. For example, if select mode, single mode, and 4-buffer mode are selected, conversion finishes when a hardware trigger has been input four times.

31.4.1.2 Operating mode

There are two operating modes: select mode and scan mode. The select mode has 1-buffer mode and 4-buffer mode as sub-modes. These modes can be set in the BS and MS bits in the ADM1 register.

(1) Select Mode

A/D conversion proceeds for the analog input specified in the CHSEL[7:0] bits in the ADM2 register. The A/D conversion results are stored in the ADCR7 to ADCR0 registers corresponding to the ADC_CH7 to ADC_CH0 pins. In select mode, 1-buffer mode or 4-buffer mode can be used as the method for storing the A/D conversion results.

Only one channel is selectable for A/D conversion in select mode.

If more than one analog input channel is selected in select mode, a channel select error occurs.

(a) 1-buffer mode

A/D conversion proceeds only once for the analog input specified in the ADM2 register and the conversion result is stored in the ADCR7 to ADCR0 registers corresponding to the ADC_CH7 to ADC_CH0 pins.

The ADC_CH7 to ADC_CH0 pins correspond with the ADCR7 to ADCR0 registers on a one-on-one basis. If interrupt output is enabled in the INTEN[7:0] bits in the ADINT register for the channels on which A/D conversion proceeds, an A/D conversion end interrupt (INTAD) occurs on completion of a single conversion.

This mode is usable for reading the result of a single conversion.

(b) 4-buffer mode

A/D conversion proceeds four times for the analog input specified in the ADM2 register and the conversion results are stored in the ADCR7 to ADCR4 or ADCR3 to ADCR0 registers. See **Table 31.7** for the correspondence between the analog inputs and A/D conversion result registers.

If interrupt output is enabled in the INTEN[7:0] bits in the ADINT register for the channels corresponding to the ADCR7 to ADCR0 registers where the A/D conversion result is to be stored in 1-buffer mode, an A/D conversion end interrupt (INTAD) occurs when the A/D conversion result is stored in the ADCR7 to ADCR0 registers. For example, to generate an interrupt on the fourth A/D conversion when ADC_CH0 is selected, set the INTEN3 bit in the ADINT register to 1.

See **Table 31.6** for the relationship between the setting of the ADINT register and the output of an A/D conversion end interrupt in 4-buffer mode.

This mode is usable for obtaining the average of the A/D conversion results.

Table 31.6 ADINT Setting and Generation of A/D Conversion End Interrupt in 4-Buffer Mode

Channel for A/D Conversion	ADINT Register Setting		A/D Conversion End Interrupt
1 channel is selected from ADC_CH3 to ADC_CH0	INTEN[0]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 0.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 0.
	INTEN[1]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 1.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 1.
	INTEN[2]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 2.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 2.
	INTEN[3]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 3.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 3.
1 channel is selected from ADC_CH7 to ADC_CH4	INTEN[4]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 4.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 4.
	INTEN[5]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 5.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 5.
	INTEN[6]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 6.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 6.
	INTEN[7]	1: Interrupt is enabled	An interrupt is output on completion of A/D conversion on channel 7.
		0: Interrupt is disabled	No interrupt is output on completion of A/D conversion on channel 7.

Table 31.7 Correspondence between Analog Inputs and A/D Conversion Result Registers in 4-Buffer Mode

Analog Input	A/D Conversion Result Register
1 channel is selected from ADC_CH3 to ADC_CH0	ADCR0 (first time)
	ADCR1 (second time)
	ADCR2 (third time)
	ADCR3 (fourth time)
1 channel is selected from ADC_CH7 to ADC_CH4	ADCR4 (first time)
	ADCR5 (second time)
	ADCR6 (third time)
	ADCR7 (fourth time)

(2) Scan Mode

A/D conversion proceeds for the analog inputs of channels selected in the ADM2 register, in ascending order of channel number. The A/D conversion results are stored in the ADCR7 to ADCR0 registers corresponding to the analog inputs. If interrupt output is enabled in the INTEN[7:0] bits in the ADINT register for the channels on which A/D conversion proceeds, an A/D conversion end interrupt (INTAD) occurs when A/D conversion on the relevant channel finishes.

This mode is usable for constantly monitoring multiple analog signals.

In scan mode, only 1-buffer mode can be specified. If 4-buffer mode is specified in scan mode, a channel select error occurs.

31.4.1.3 Conversion Mode

Single mode and repeat mode are available. Use the RPS bit in the ADM1 register to set the conversion mode.

(1) Single Mode

When A/D conversion for the number of times of the conversion count finishes, the ADCE bit in the ADM0 register is automatically cleared to 0. For the number of times of conversion, see the description of operation in **Table 31.5**.

(2) Repeat Mode

When 1 is written to the ADCE bit in the ADM0 register, A/D conversion repeatedly proceeds in the A/D conversion mode specified in the ADM1 register. Immediately after 0 is written to the ADCE bit in the ADM0 register, A/D conversion is finished.

For the number of times of conversion, see the description of operation in **Table 31.5**.

31.4.2 Interrupt Functions

Table 31.8 lists the interrupt sources of the A/D converter and the conditions for generating them and the methods for confirming and clearing the interrupt sources.

The INTS bit in the ADINT register is used to set the operation for outputting an interrupt request of the INTAD.

Table 31.8 List of Interrupt Functions

Interrupt Source	Condition for Generation	Enabling Interrupt	Confirming Interrupt Source	Method for Clearing Interrupt Source
		ADINT Register	ADSTS Register*[1]	ADSTS Register
A/D conversion end	When A/D conversion on the specified channel is completed	INTEN[7:0]*[2]	INTST[7:0]	Write 1 to the bits that were 1 when the register was read.
A/D conversion channel select error	● When all of the CHSEL[7:0] bits in the ADM2 register were 0 at the start of A/D conversion*[3] ● When multiple analog input channels are selected in select mode*[3] ● When 4-buffer mode is specified in scan mode*[3]	CSEEN	CSEST	Write 1 to the CSEST bit.

Note 1. If an interrupt source is generated regardless of the settings of the INTEN[7:0] and CSEEN bits in the ADINT register, the bit corresponding to the interrupt source is set to 1 in the ADSTS register.

Note 2. The A/D conversion end interrupt can be enabled or disabled individually for each channel.

Note 3. When an A/D conversion channel select error occurs, the ADCE bit in the ADM0 register becomes 0 and A/D conversion operation is stopped.

31.4.3 Procedures of A/D Conversion

Follow the procedures given below for A/D conversion.

(1) Procedure for starting A/D conversion

After the A/D converter is released from the software reset state, use the ADM3 to ADM0 registers*[1] to select the analog input channels on which A/D conversion proceeds and specify the trigger mode (software trigger or hardware trigger), operating mode (select or scan), etc.

Setting the ADCE bit in the ADM0 register to 1 will start A/D conversion in software trigger mode, whereas the A/D converter will enter a standby state for a trigger*[2] in hardware trigger mode.

Note 1. Set the ADM3 to ADM0 registers when conversion by the A/D converter is stopped (ADBSY bit in the ADM0 register is 0).

Note 2. If the ADCE bit in the ADM0 register is set to 1 in hardware trigger mode, a transition is made to a trigger standby state. A/D conversion is started (ADBSY bit in ADM0 register is 1) by a hardware trigger signal and the A/D converter returns to the trigger standby state (ADBSY bit in ADM0 register is 0) on completion of A/D conversion.

A/D conversion operation is enabled.

After the A/D converter makes a transition from power-saving mode to normal mode, a period as long as the stabilization wait time is required. The stabilization wait time is 1 μs.

After the stabilization wait time has passed, set the ADCE bit in the ADM0 register to 1.

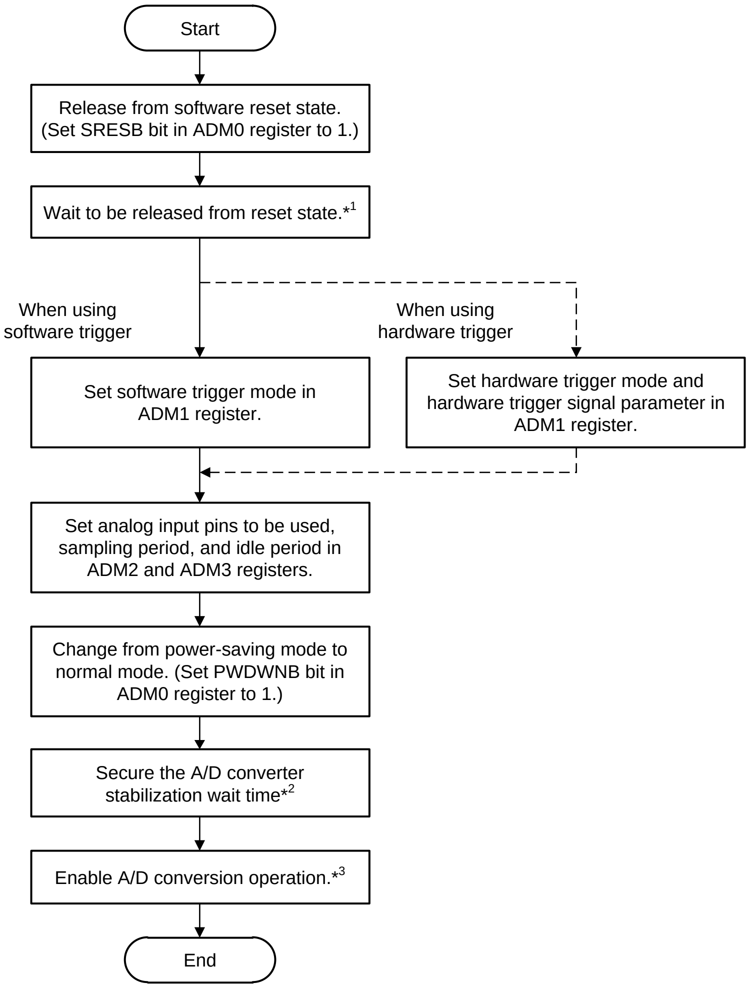
In software trigger mode, setting the ADCE bit to 1 starts A/D conversion.

In hardware trigger mode, setting the ADCE bit to 1 makes the A/D converter enter a standby state for a hardware trigger after 250 ns has passed, and conversion operation is started upon detection of a valid edge.

(2) When A/D conversion is finished, the A/D conversion results are stored in the ADCR7 to ADCR0 registers. Also, an A/D conversion end interrupt (INTAD) occurs on completion of A/D conversion on the channels specified in the INTEN[7:0] bits in the ADINT register.

31.4.3.1 Procedure for Starting A/D Conversion

Use the following flowchart to start A/D conversion.



Note 1. The time waiting for the A/D converter to be released from the reset state is 2 × ADC_ADCLK(TSUϕ).

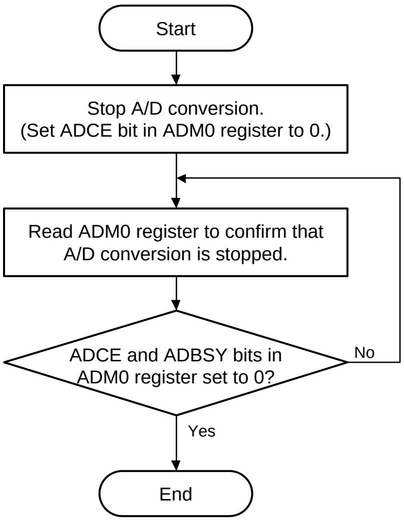
Note 2. The stabilization wait time is 1 µs.

Note 3. In software trigger mode, A/D conversion is started by setting the ADCE bit to 1.
In hardware trigger mode, after the ADCE bit is set to 1, A/D conversion is started upon detection of a valid edge.

Figure 31.4 Procedure for Starting A/D Conversion

31.4.3.2 Procedure for Stopping A/D Conversion

Use the following flowchart to stop A/D conversion.

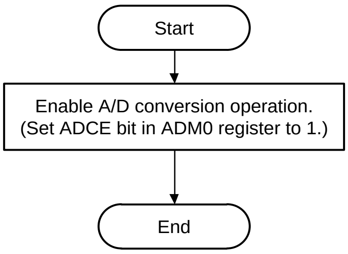


Note: For details on the operation for stopping conversion by writing 0 to the ADCE bit, see **Table 31.10**.

Figure 31.5 Procedure for Stopping A/D Conversion

31.4.3.3 Procedure for Stopping and Restarting A/D Conversion

Use the following flowchart to restart A/D conversion after it has been stopped with the procedure for stopping A/D conversion.



Note: For details on the operation for restarting conversion, see **Table 31.10**.

Figure 31.6 Procedure for Stopping and Restarting A/D Conversion

31.4.3.4 Procedure for Entering Power-saving Mode

Use the following flowchart to make a transition to the power-saving mode.

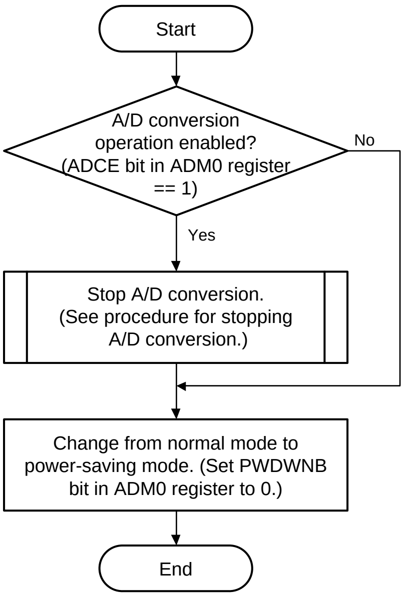
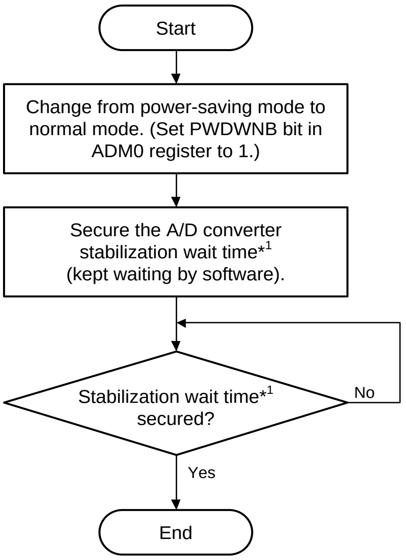


Figure 31.7 Procedure for Entering Power-Saving Mode

31.4.3.5 Procedure for Releasing from Power-saving Mode

Use the following flowchart to release the A/D converter from the power-saving mode.



Note 1. The stabilization wait time is 1 µs.

Figure 31.8 Procedure for Releasing from Power-Saving Mode

31.4.3.6 Procedure for a Software Reset

Use the following flowchart to execute a software reset.

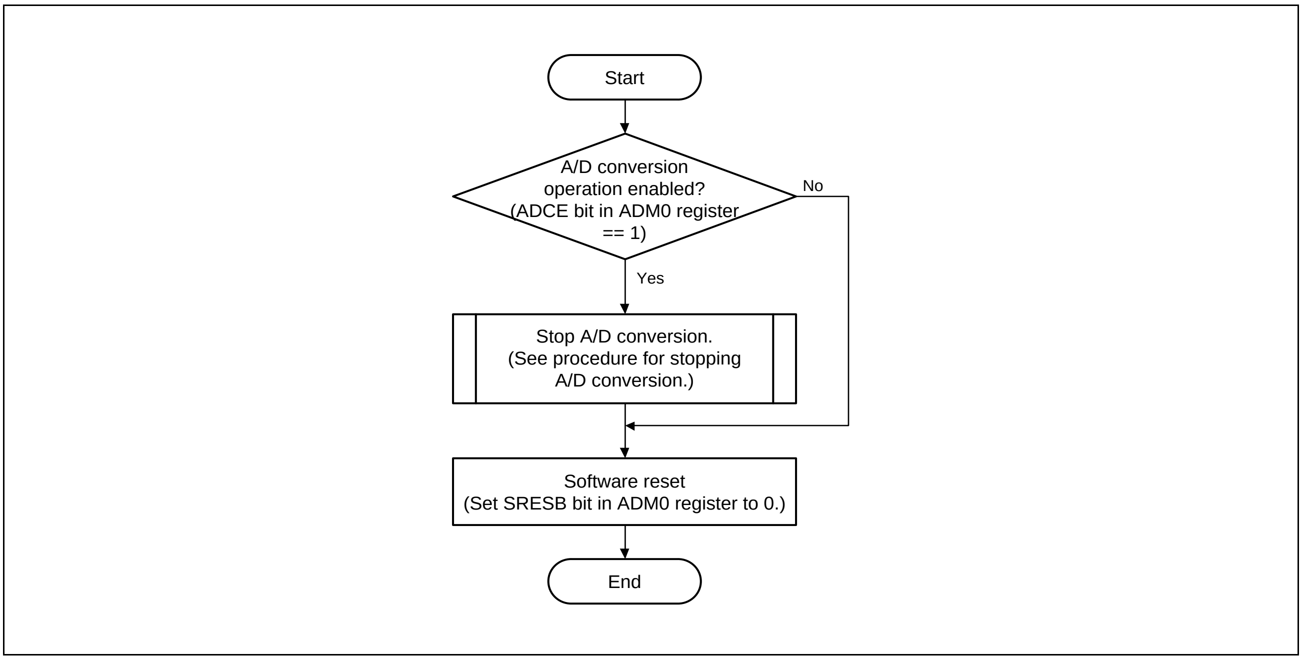
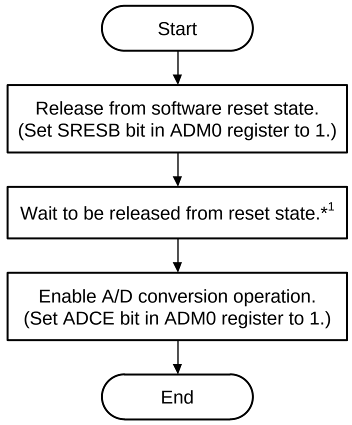


Figure 31.9 Procedure for a Software Reset

31.4.3.7 Procedure for Restarting A/D Conversion after a Software Reset

Use the following flowchart to start A/D conversion, then execute a software reset with the procedure for a software reset, and finally restart A/D conversion.



Note 1. The time waiting for the A/D converter to be released from the reset state is 2 × ADC_ADCLK(TSUϕ).

Figure 31.10 Procedure for Restarting A/D Conversion after a Software Reset

31.4.3.8 Procedure for Interrupt Processing

Use the flowchart in **Figure 31.11** to process an interrupt request that was output by the A/D converter.

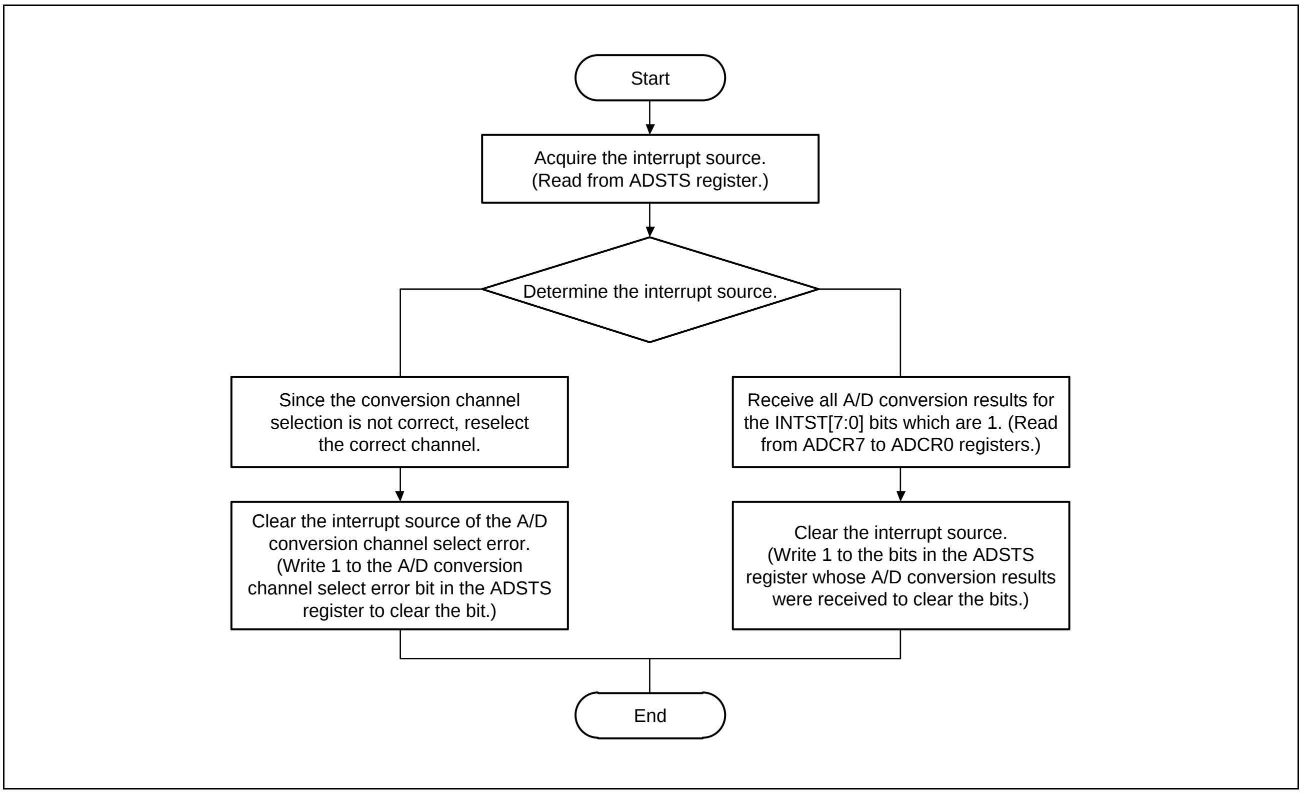


Figure 31.11 Procedure for Interrupt Processing

31.4.4 Examples of A/D Conversion

This section gives examples of A/D conversion.

1) Example of A/D conversion in select mode with single mode selected
2) Example of A/D conversion in select mode with repeat mode selected
3) Example of A/D conversion in 4-buffer mode
4) Example of A/D conversion in scan mode with single mode selected
5) Example of A/D conversion in scan mode with repeat mode selected
6) Example of A/D conversion in auto mode
7) Example of A/D conversion in step mode

Note to confirm the differences between modes shown below because not all possible combinations are included in this manual.

- For the differences between 1-buffer mode and 4-buffer mode, refer to 1) and 3).
- For the differences between single mode and repeat mode, refer to 1) and 2).
- For the differences between select mode and scan mode, refer to 1) and 4) or 2) and 5)
- For the differences between auto mode and step mode, refer to 6) and 7)
- For the differences between software trigger mode and hardware trigger mode, refer to 4) and 6).

31.4.4.1 Example of A/D Conversion in Select Mode with Single Mode Selected

Figure 31.12 shows an example of A/D conversion with software trigger mode, select mode, single mode and 1-buffer mode are selected. In the example in **Figure 31.12**, conversion proceeds with the analog input changed from ADC_CH1 to ADC_CH2.

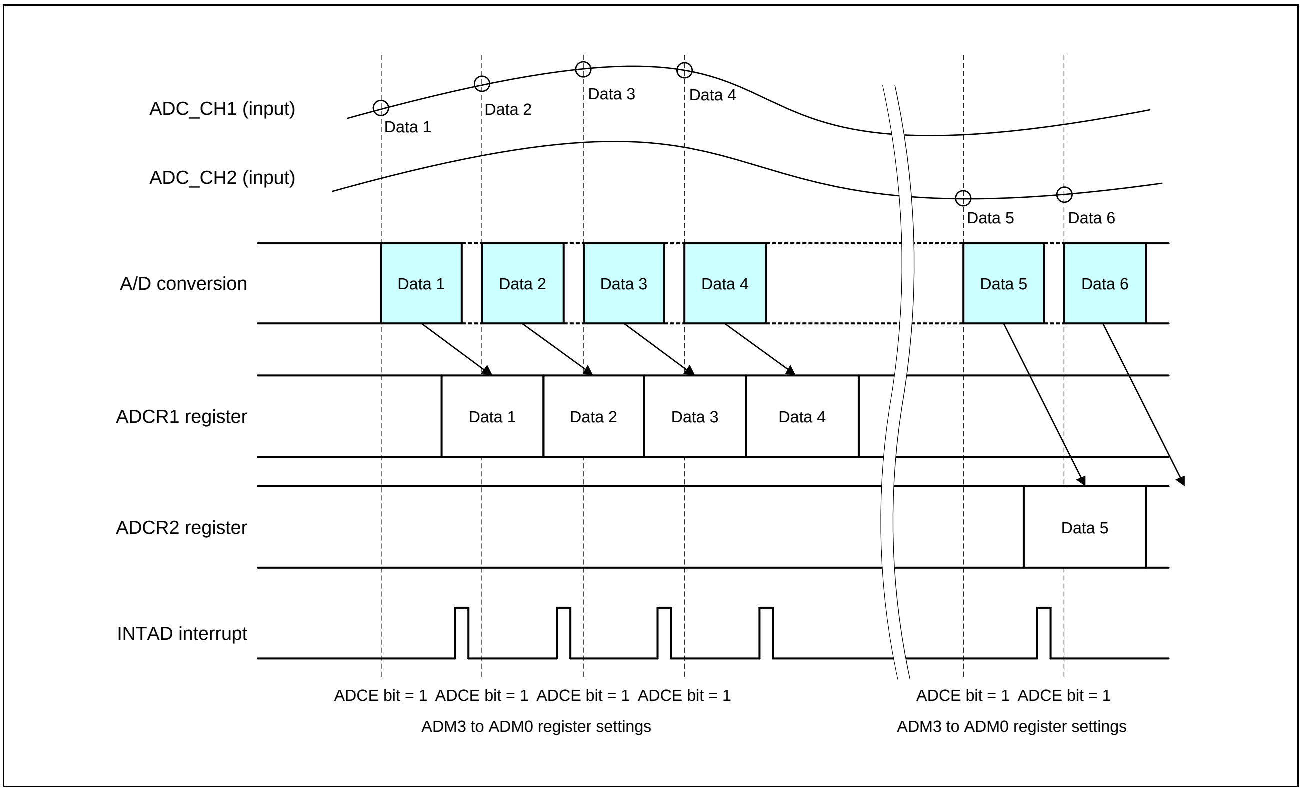


Figure 31.12 Example of A/D Conversion in Select Mode with Single Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, select mode, single mode, and 1-buffer mode are selected, and analog input pin ADC_CH1 is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 1 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH1 (A/D conversion result is stored in the ADCR1 register).
9. An INTAD interrupt occurs.
10. Steps 7 to 9 above are repeated.
11. A/D conversion operation is stopped by setting the ADCE bit in the ADM0 register to 0.
12. In the ADM3 to ADM0 registers, software trigger mode, select mode, single mode, and 1-buffer mode are selected, and analog input pin ADC_CH2 is set.

13. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
14. A/D conversion proceeds on ADC_CH2 (A/D conversion result is stored in the ADCR2 register).
15. An INTAD interrupt occurs.
16. Steps 13 to 15 above are repeated.

Note 1. The stabilization wait time is 1 μs.

31.4.4.2 Example of A/D Conversion in Select Mode with Repeat Mode Selected

Figure 31.13 shows an example of A/D conversion with software trigger mode, select mode, repeat mode, and 1-buffer mode are selected. In the example in **Figure 31.13**, conversion proceeds with the analog input changed from ADC_CH1 to ADC_CH2.

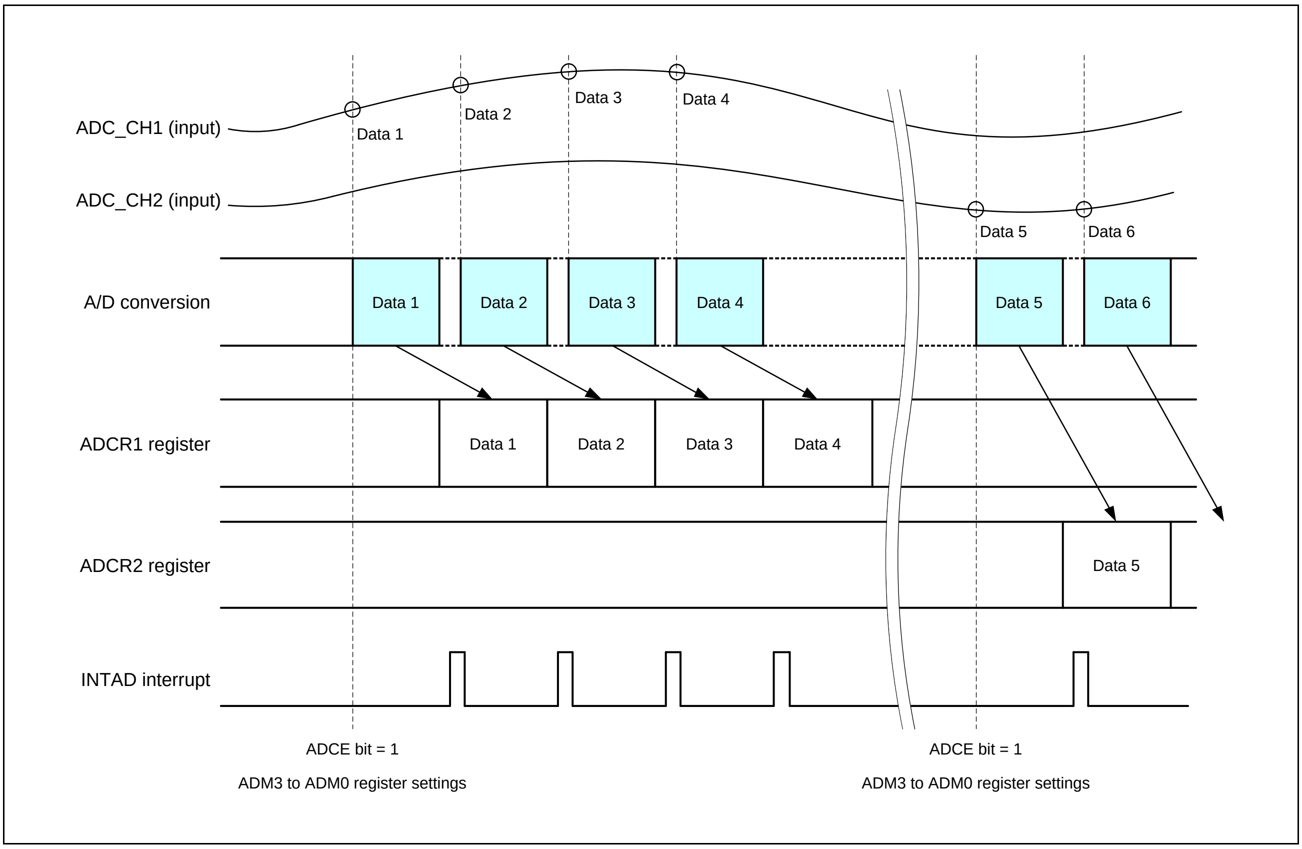


Figure 31.13 Example of A/D Conversion in Select Mode with Repeat Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, and 1-buffer mode are selected, and analog input pin ADC_CH1 is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 1 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH1 (A/D conversion result is stored in the ADCR1 register).
9. An INTAD interrupt occurs.
10. Steps 8 and 9 above are repeated automatically.
11. A/D conversion operation is stopped by setting the ADCE bit in the ADM0 register to 0.

12. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, 1-buffer mode are selected, and analog input pin ADC_CH2 is set.
13. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
14. A/D conversion proceeds on ADC_CH2 (A/D conversion result is stored in the ADCR2 register).
15. An INTAD interrupt occurs.
16. Steps 14 and 15 above are repeated.

Note 1. The stabilization wait time is 1 µs.

31.4.4.3 Example of A/D Conversion in 4-Buffer Mode

Figure 31.14 shows an example of A/D conversion with software trigger mode, select mode, repeat mode, and 4-buffer mode are selected. In the example in **Figure 31.14**, conversion proceeds with the analog input changed from ADC_CH2 to ADC_CH4.

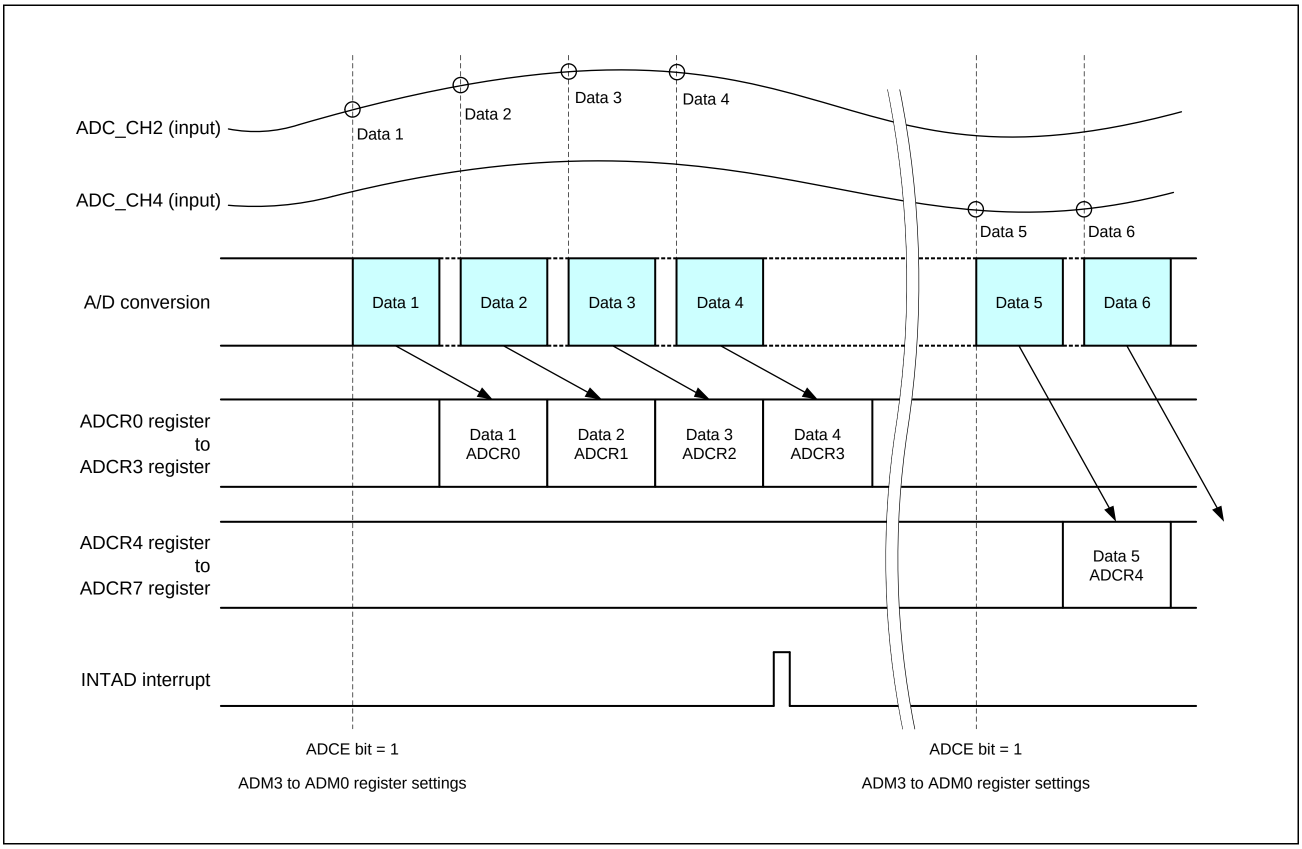


Figure 31.14 Example of A/D Conversion in 4-Buffer Mode

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, and 4-buffer mode are selected, and analog input pin ADC_CH2 is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 1) is stored in the ADCR0 register).
9. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 2) is stored in the ADCR1 register).
10. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).
11. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 4) is stored in the ADCR3 register).

12. An INTAD interrupt occurs.
13. Steps 8 to 12 above are repeated automatically.
14. A/D conversion operation is stopped by setting the ADCE bit in the ADM0 register to 0.
15. In the ADM3 to ADM0 registers, software trigger mode, select mode, repeat mode, and 4-buffer mode are selected, and analog input pin ADC_CH4 is set.
16. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
17. A/D conversion proceeds on ADC_CH4 (A/D conversion result (data 5) is stored in the ADCR4 register).
18. A/D conversion proceeds on ADC_CH4 (A/D conversion result (data 6) is stored in the ADCR5 register).

Note 1. The stabilization wait time is 1 µs.

31.4.4.4 Example of A/D Conversion in Scan Mode with Single Mode Selected

Figure 31.15 shows an example of A/D conversion with software trigger mode, scan mode, and single mode are selected. In the example in **Figure 31.15**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

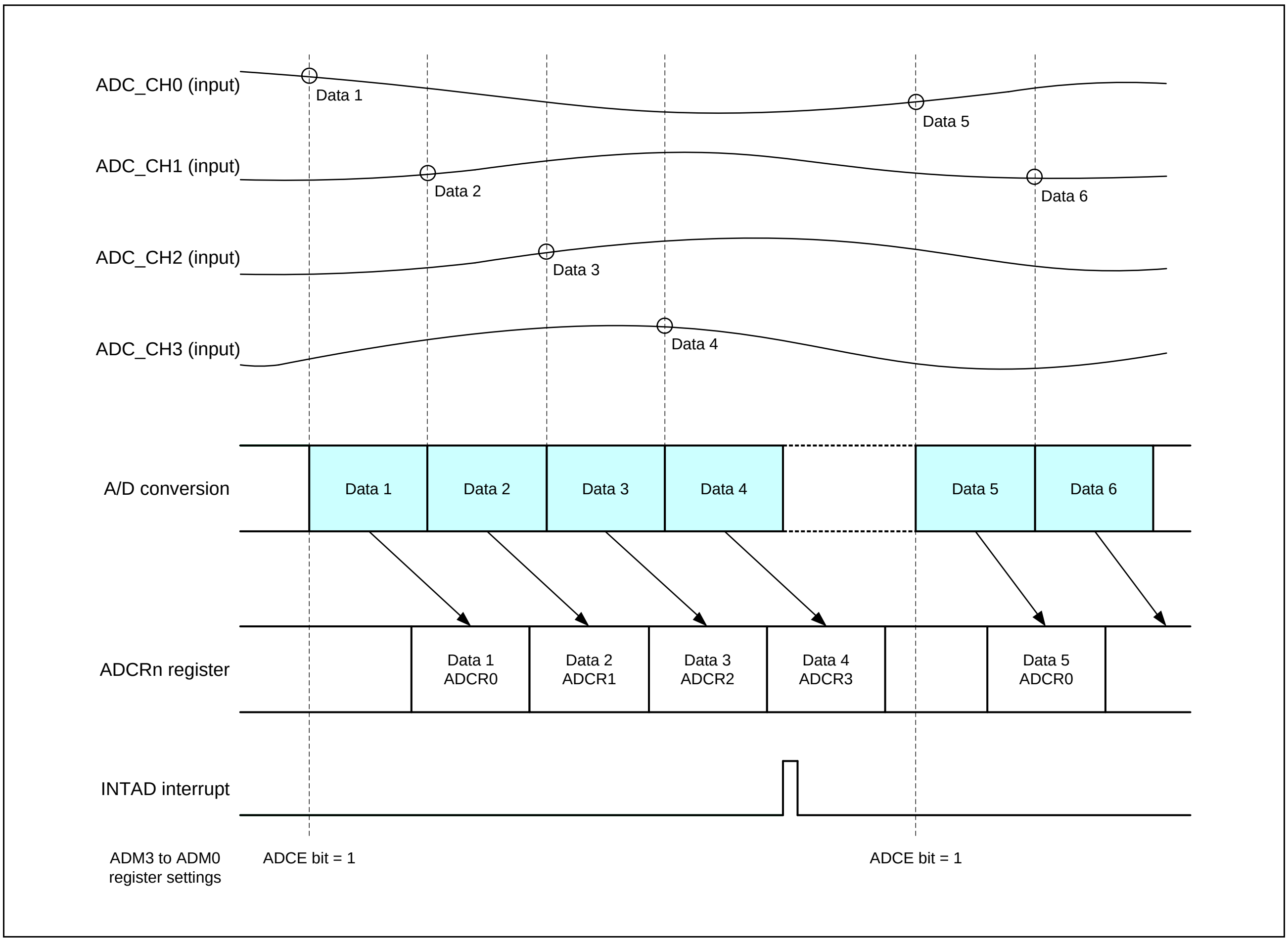


Figure 31.15 Example of A/D Conversion in Scan Mode with Single Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, scan mode, and single mode are selected, and analog input pins ADC_CH3 to ADC_CH0 are set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
9. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
10. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).

11. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
12. An INTAD interrupt occurs.
13. Steps 7 to 12 above are repeated.

Note 1. The stabilization wait time is 1 µs.

31.4.4.5 Example of A/D Conversion in Scan Mode with Repeat Mode Selected

Figure 31.16 shows an example of A/D conversion with software trigger mode, scan mode, and repeat mode are selected. In the example in **Figure 31.16**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

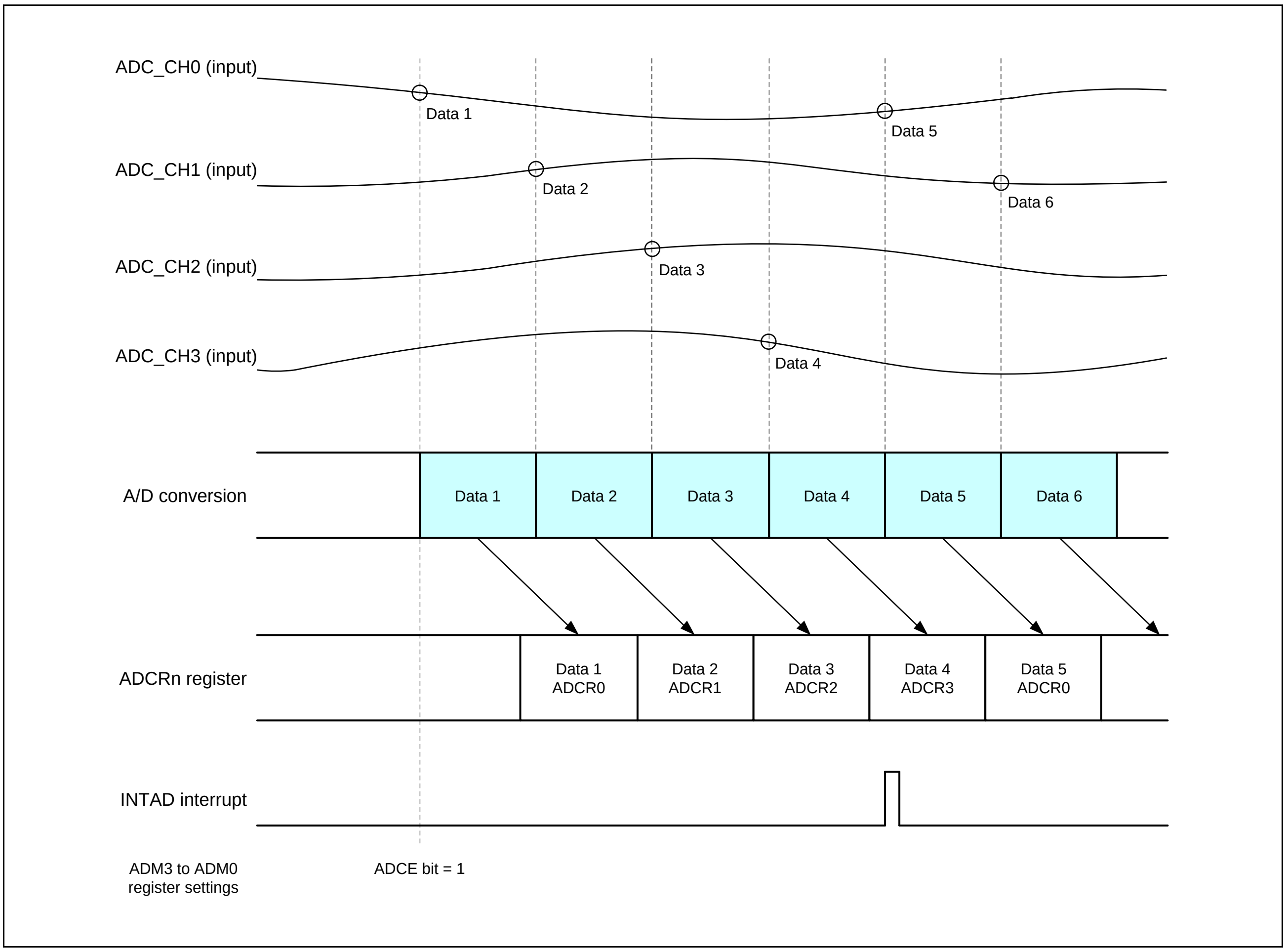


Figure 31.16 Example of A/D Conversion in Scan Mode with Repeat Mode Selected

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, software trigger mode, scan mode, and repeat mode are selected and analog input pins ADC_CH3 to ADC_CH0 are set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).
8. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
9. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
10. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).

11. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
12. An INTAD interrupt occurs.
13. Steps 8 to 12 above are repeated automatically.

Note 1. The stabilization wait time is 1 µs.

31.4.4.6 Example of A/D Conversion in Auto Mode

Figure 31.17 shows an example of A/D conversion with hardware trigger mode, auto mode, scan mode, and single mode are selected. In the example in **Figure 31.17**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

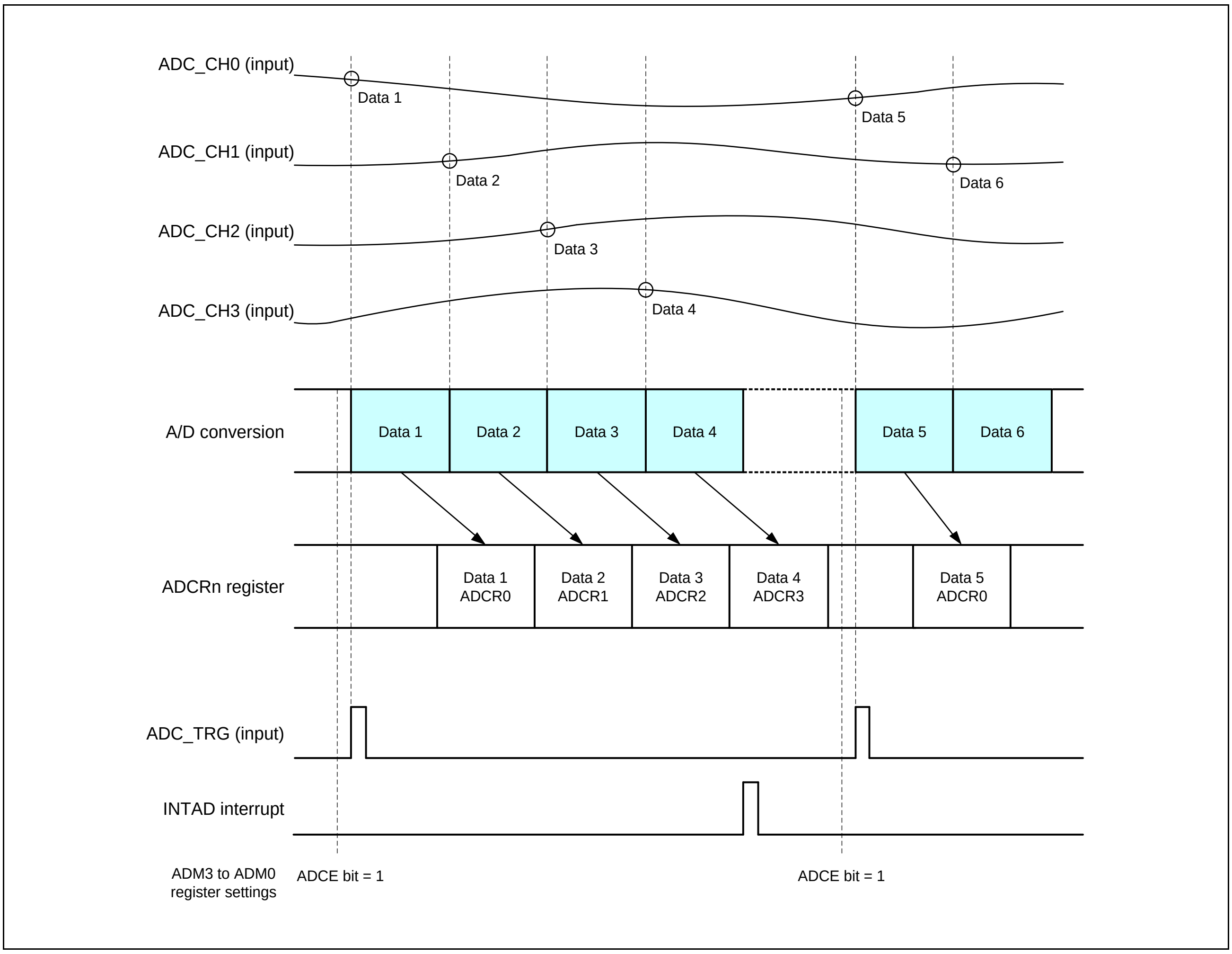


Figure 31.17 Example of A/D Conversion in Auto Mode

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, hardware trigger mode, auto mode, scan mode, and single mode are selected, and analog input pins ADC_CH3 to ADC_CH0 are set, the trigger of the ADC_TRG pin is enabled, and rising-edge detection of the hardware trigger is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1 (A/D conversion is started).

8. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
9. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
10. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
11. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).
12. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
13. An INTAD interrupt occurs.
14. Steps 7 to 13 above are repeated.

Note 1. The stabilization wait time is 1 µs.

31.4.4.7 Example of A/D Conversion in Step Mode

Figure 31.18 shows an example of A/D conversion with hardware trigger mode, step mode, scan mode, and single mode are selected. In the example in **Figure 31.18**, conversion proceeds with ADC_CH3 to ADC_CH0 selected.

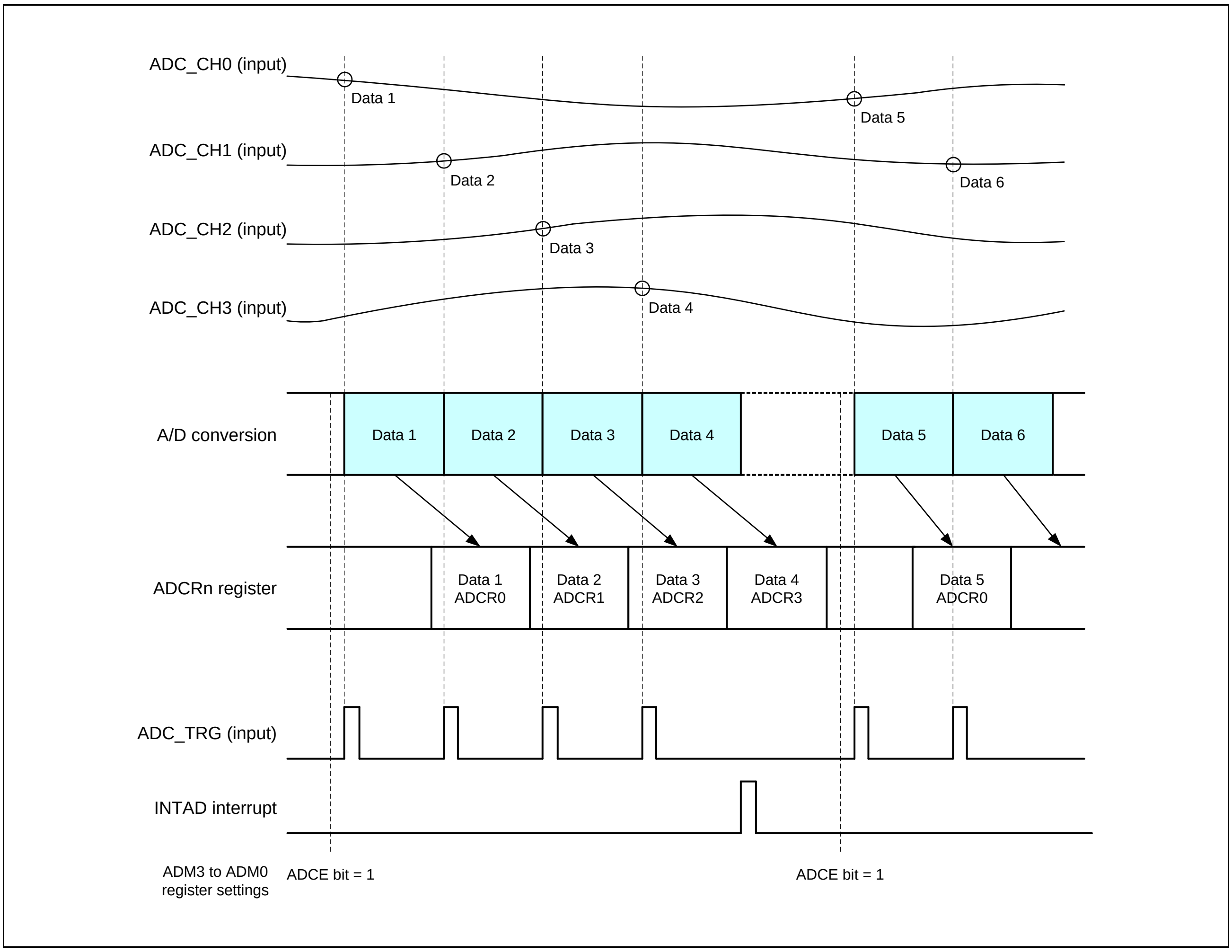


Figure 31.18 Example of A/D Conversion in Step Mode

1. The SRESB bit in the ADM0 register is set to 1 to release the A/D converter from the software reset state.
2. The A/D converter waits to be released from the reset state (wait time is 2 × ADC_ADCLK(TSUϕ)).
3. In the ADM3 to ADM0 registers, hardware trigger mode, step mode, scan mode, and single mode are selected, analog input pins ADC_CH3 to ADC_CH0 are set, the trigger of the ADC_TRG pin is enabled, and rising-edge detection of the hardware trigger is set.
4. In the ADINT and ADSTS registers, the interrupt signal is set as a pulse and interrupt output of channel 3 is enabled.
5. Normal mode is selected by setting the PWDWNB bit in the ADM0 register to 1.
6. The A/D converter is kept waiting for a period equal to or longer than the stabilization wait time*[1] by software.
7. A/D conversion operation is enabled by setting the ADCE bit in the ADM0 register to 1.
8. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).

9. A/D conversion proceeds on ADC_CH0 (A/D conversion result (data 1) is stored in the ADCR0 register).
10. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
11. A/D conversion proceeds on ADC_CH1 (A/D conversion result (data 2) is stored in the ADCR1 register).
12. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
13. A/D conversion proceeds on ADC_CH2 (A/D conversion result (data 3) is stored in the ADCR2 register).
14. After 250 ns that is the period it takes for accepting a trigger, a rising edge of the pulse is input to the ADC_TRG pin (A/D conversion is started).
15. A/D conversion proceeds on ADC_CH3 (A/D conversion result (data 4) is stored in the ADCR3 register).
16. An INTAD interrupt occurs.
17. Steps 7 to 16 above are repeated.

Note 1. The stabilization wait time is 1 µs.

31.5 Notes

31.5.1 Interval of Hardware Trigger

The interval (interval of input time) of a trigger in hardware trigger mode must be longer than the A/D conversion time. For the A/D conversion time, see **Section 31.3.4, A/D Converter Mode Register 3 (ADM3)**.

31.5.2 Timing Restrictions

This A/D converter has restrictions on the timing interval, which are caused by the asynchronism countermeasure circuit. The restrictions shown in **Table 31.9** need to be met. If an access is made in a period shorter than this interval, the register values on the ADC_PCLK(P0ϕ) side are not reflected in the registers on the ADC_ADCLK(TSUϕ) side because the notification of register update does not reach the ADC_ADCLK(TSUϕ) side.

Table 31.9 Restrictions on Timing Interval

Condition	Restriction
Writing to the same register successively	(Interval of write access) ≥ (6 × ADC_PCLK(P0ϕ) + 6 × ADC_ADCLK(TSUϕ))
Interval of A/D conversion	(A/D conversion time) ≥ (6 × ADC_PCLK(P0ϕ) + 6 × ADC_ADCLK(TSUϕ))

Note: For the A/D conversion time, see **Figure 31.2**.

31.5.3 Restrictions on ADC_PCLK(P0ϕ) and ADC_ADCLK(TSUϕ) Cycles

The ADC_PCLK(P0ϕ) and ADC_ADCLK(TSUϕ) cycles need to satisfy the relation in the following formula.

$$\frac{\mathrm{ADC\_ADCLK(TSU\phi)} \le (\mathrm{A/D\ conversion\ time})}{6 - \mathrm{ADC\_PCLK(P0\phi)}}$$

Note: For the A/D conversion time, see **Figure 31.2**.

31.5.4 Operation when Stopping or Restarting A/D Conversion

When 0 is written to the ADCE bit in the ADM0 register, the A/D converter stops A/D conversion. A/D conversion is restarted by writing 0 to the ADCE bit in the ADM0 register once and then writing 1 to the same bit. Even though there was an instruction to stop A/D conversion, A/D conversion may not stop immediately. **Table 31.10** shows the operation for stopping conversion and also the operation for restarting conversion after it has been stopped.

Table 31.10 Operation when Stopping or Restarting A/D Conversion with the ADCE Bit

A/D Conversion Modes					Timing for Stopping Conversion when Stop is Instructed with ADCE = 0	Operation for Restarting Conversion
Trigger		Operating Mode	Conversion Count	Buffer Count		
Mode	Input					
Software	—	Select	Single	1	Stopped on completion of 1-time conversion.	Conversion proceeds on the channel selected in the ADM2 register.
				4	Stopped on completion of conversion in which there was an instruction to stop in the middle of 4-time conversion.	Same as above
			Repeat	1	Stopped on completion of 1-time conversion.	Same as above
				4	Stopped on completion of conversion in which there was an instruction to stop in the middle of 4-time conversion.	Same as above
		Scan	Single	1	Stopped on completion of conversion in which there was an instruction to stop in the middle of scanning.	Conversion proceeds from the smallest channel number selected in the ADM2 register.
				4	(Setting prohibited)	(Setting prohibited)
			Repeat	1	Stopped on completion of conversion in which there was an instruction to stop in the middle of scanning.	Conversion proceeds from the smallest channel number selected in the ADM2 register.
				4	(Setting prohibited)	(Setting prohibited)
Hardware	Auto mode	Select	Single	1	Same as software trigger mode	Same as software trigger mode
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above
		Scan	Single	1	Same as above	Same as above
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above
	Step mode	Select	Single	1	Same as software trigger mode	Same as software trigger mode
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above
		Scan	Single	1	Same as above	Same as above
				4	Same as above	Same as above
			Repeat	1	Same as above	Same as above
				4	Same as above	Same as above

32. USB2.0

This LSI includes 1 channel USB2.0 OTG/DRD(Host/Function) interface and 1 channel USB2.0 Host interface.

This section describes the overview of USB2.0 and USB2.0 PHY control.

The detail function of USB2.0 Host controller and the common function both Host module and Function module are described in **Section 32A, USB 2.0 Host Module**.

The detail function of USB2.0 Function controller is described in **Section 32B, USB 2.0 Function Module**.

32.1 Features

This interface complies the following specifications.

- Universal Serial Bus Specification Revision 2.0
- On-The-Go and Embedded Host Supplement to the USB Revision 2.0 Specification Revision 2.0 plus errata and ecn*[1]
- Enhanced Host Controller Interface (EHCI) Specification for Universal Serial Bus Revision 1.0
- EHCI v1.1 Addendum*[2]
- Open Host Controller Interface (OHCI) Specification for USB Release 1.0a
- Battery Charging Specification Revision 1.2*[3]

Note 1. Session Request Protocol(SRP) and Host Negotiation Protocol(HNP) are not supported.
Note 2. Some EHCI v1.1 feature are not supported. (Refer to **Section 32A, USB 2.0 Host Module** for detail)
Note 3. DCP (Dedicated Charging Port) is not supported as downstream port.

32.1.1 Ch0: USB2.0 OTG/DRD(Host/Function) interface

The following table shows features of this interface.

Table 32.1 Ch0: USB2.0 OTG/DRD(Host/Function) interface features

Function	Description
Host mode	● Support mode: High-Speed(480Mbps)/Full-Speed(12Mbps)/Low-Speed(1.5Mbps) ● Support Isochronous/Interrupt/Control/Bulk transfer modes ● Support Isochroous/Interrupt high bandwidth transfer
Function mode	● Support mode: High-Speed(480Mbps)/Full-Speed(12Mbps) ● Isochronous/Interrupt/Control/Bulk transfer ● Up to 10 ch PIPE (includes default control PIPE)
Other functions	● OTG function (Rev2.0) ● Battery Charging function ● DRD(Dual-Role-Device) function (Static switch between Host and Function)

32.1.2 Ch1: USB2.0 HOST interface

The following table shows features of this interface.

Table 32.2 Ch1: USB2.0 Host interface function features

Features	Description
Host mode	● Support mode: High-Speed(480Mbps)/Full-Speed(12Mbps)/Low-Speed(1.5Mbps) ● Support Isochronous/Interrupt/Control/Bulk transfer modes ● Support Isochroous/Interrupt high bandwidth transfer
Other function	● Battery Charging function

32.1.3 Block Diagram

The block diagram of USB interface is as follows.

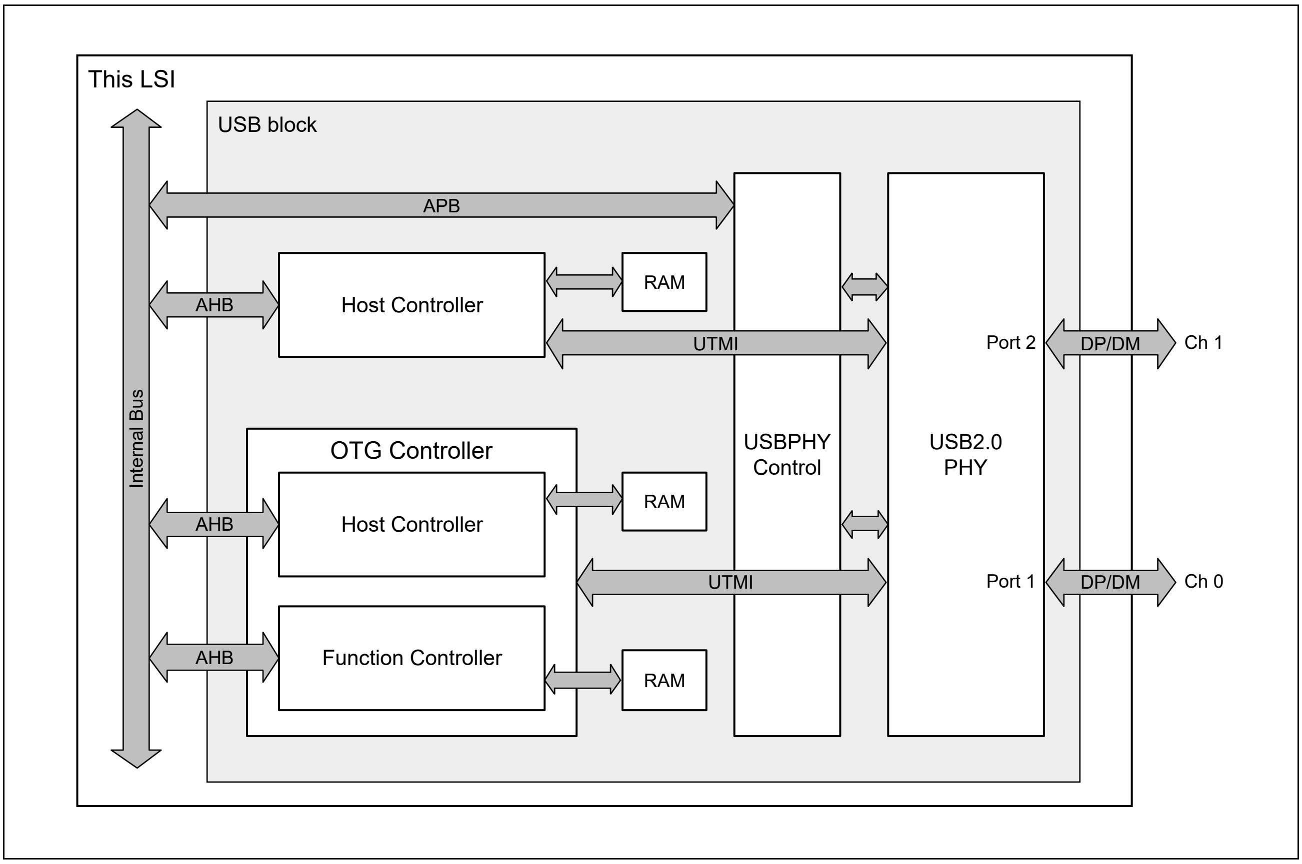


Figure 32.1 USB interface block diagram

32.1.4 External signal pins

The following table shows external signal pins for USB block.

Table 32.3 External signal pins

Name	I/O	Description	Active Level	Initial value	Handling in not used
USB0_DP	—	USB data pin Data+ (Ch0)	—	—	Open*[1]
USB0_DM	—	USB data pin Data- (Ch0)	—	—	Open*[1]
USB0_OVRCUR	I	Over current detection input (Ch0)	L	—	Pull-Up
USB0_VBUSEN	O	VBUS output enable (Ch0)	H	0b	Open
USB0_VBUSIN	I	Peripheral VBUS input (Ch0)	H	—	Pull-Down
USB0_OTG_EXICEN	O	External power IC control (Ch0)	H	0b	Open
USB0_OTG_ID	I	OTG ID input (Ch0)	—	—	Pull-Down
USB1_DP	—	USB data pin Data+ (Ch1)	—	—	Open*[1]
USB1_DM	—	USB data pin Data- (Ch1)	—	—	Open*[1]
USB1_OVRCUR	I	Over current detection input (Ch1)	L	—	Pull-Up
USB1_VBUSEN	O	VBUS output enable (Ch1)	H	0b	Open

Note 1. When it is not used permanently, connect to GND via a resistor of 10kΩ.

32.1.5 Power and GND pins

The following table shows Power and GND pins for USB block.

Table 32.4 Power and GND pins

Name	Description
USB_VDD18	Internal Regulator Power (1.8V), HS receiver power (1.8V)
USB_VDD33	IO power (3.3V)
USB_RREF	Reference voltage
VSS	GND

32.1.6 Interrupt

The following table shows interrupt list for USB block.

Table 32.5 USB interrupt

Name	Description	Active Level	Type
U2H0_INT	USB2.0 Host AHB Interrupt (Ch0)	H	Level-Sensitive
U2H0_OHCI_INT	USB2.0 Host OHCI Interrupt (Ch0)	H	Level-Sensitive
U2H0_EHCI_INT	USB2.0 Host EHCI Interrupt (Ch0)	H	Level-Sensitive
U2H0_WAKEON_INT	USB2.0 Host EHCI Wakeup Interrupt (Ch0)	H	Level-Sensitive
U2H0_OBINT	USB2.0 Host OTG and Battery Charging Interrupt (Ch0)	H	Level-Sensitive
U2H1_INT	USB2.0 Host AHB Interrupt (Ch1)	H	Level-Sensitive
U2H1_OHCI_INT	USB2.0 Host OHCI Interrupt (Ch1)	H	Level-Sensitive
U2H1_EHCI_INT	USB2.0 Host EHCI Interrupt (Ch1)	H	Level-Sensitive
U2H1_WAKEON_INT	USB2.0 Host EHCI Wakeup Interrupt (Ch1)	H	Level-Sensitive
U2H1_OBINT	USB2.0 Host OTG and Battery Charging Interrupt (Ch1)	H	Level-Sensitive
U2P_IXL_INT	USB2.0 Function controller Interrupt (Ch0)	H	Edge-Triggered
U2P_INT_DMA[1:0]	USB2.0 Function controller DMA transaction complete interrupt (Ch0)	H	Level-Sensitive
U2P_INT_DMAERR	USB2.0 Function controller DMA error response interrupt (Ch0)	H	Level-Sensitive

32.2 Register Configuration

32.2.1 Host controller Register

Refer to the **Section 32A, USB 2.0 Host Module**.

32.2.2 Function controller Register

Refer to the **Section 32B, USB 2.0 Function Module**.

32.2.3 USBPHY Control Register

USBPHY Control Register mainly controls reset and power down of the USB/PHY.

Base Address: H’0_11C4_0000 (Cortex-A55 Address Space)

Base Address: H’41C4_0000 (Cortex-M33 Address Space Non-Secure)

Base Address: H’51C4_0000 (Cortex-M33 Address Space Secure)

The following table shows USBPHY Control Register list.

Prohibit to write undefined area.

Table 32.6 USBPHY Control Register List

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Reset register	RESET	R/W	H’0000_0133	H’000	32
Clock control register	UCLKCTL	R/W	H’0340_0303	H’018	32
Direct power down setting register	UDIRPD	R/W	H’0000_0000	H’01C	32
VBUSEN control register	VBENCTL	R/W	H’0000_0000	H’03C	32
Connection control register	CON_CTRL	R/W	H’0000_0000	H’020	32
Clock status register	CLK_STAT	R	H’0000_0000	H’104	32

32.3 Register Descriptions

32.3.1 Reset register (RESET)

This register controls USB/PHY reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	sel_pll reset	—	—	—	pllreset	—	—	sel_p2 reset	sel_p1 reset	—	—	phyrst_2	phyrst_1
Initial Value	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1	1
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	sel_pllreset	0	R/W	Select USB/PHY PLL reset 1: register (by bit8: pllreset) 0: control by host controller (Ch0)
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	pllreset	1	R/W	PLL reset 1: PLL reset 0: deassert PLL reset
7, 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	sel_p2reset	1	R/W	Select USB/PHY Port2 reset 1: register(phyrst_2) 0: Control by host controller (Ch1)
4	sel_p1reset	1	R/W	Select USB/PHY Port1 reset 1: register(phyrst_1) 0: Control by host controller (Ch0)
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	phyrst_2	1	R/W	USB/PHY reset (Port2) 1: USB/PHY reset 0: deassert USB/PHY reset Only when sel_p2reset=1, this bit is valid.
0	phyrst_1	1	R/W	USB/PHY reset (Port1) 1: USB/PHY reset 0: deassert USB/PHY reset Only when sel_p1reset=1, this bit is valid.

32.3.2 Clock control register (UCLKCTL)

This register controls USB/PHY clock.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	sel_ sleepm	—	—	sleepm_ 2	sleepm_ 1	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	sel_sus pendm	—	—	suspend m_2	suspend m_1	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	sel_sleepm	0	R/W	Select SLEEPM control 1: register (controlled by bit 24: sleepm_1 and bit 25: sleepm_2) 0: Control by controller
27, 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	sleepm_2	1	R/W	Sleep control (Port2) 1: Normal mode 0: Sleep mode
24	sleepm_1	1	R/W	Sleep control (Port1) 1: Normal mode 0: Sleep mode
23	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	—	1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21, 20	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19, 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17, 16	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15, 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	sel_suspendm	0	R/W	Select SUSPENDM control 1: register (controlled by bit 8: suspendm_1 and bit 9: suspendm_2) 0: control by controller
11, 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	suspend_2	1	R/W	Suspend control (Port2) 0: Suspend mode 1: Normal mode
8	suspend_1	1	R/W	Suspend control (Port1) 0: Suspend mode 1: Normal mode
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	—	All 1	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

32.3.3 Direct power down setting register (UDIRPD)

This register controls USB/PHY direct power down.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	sel_udirpd	—	—	—	dirpd
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	sel_udirpd	0	R/W	Select dirpd control 1: register (dirpd) 0: Control by host controller (Ch0)
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	dirpd	0	R/W	Power down control to reduce the power consumption of the USB/PHY when not using the USB function. 0: Normal operation 1: Power save

32.3.4 VBUSEN control register (VBENCTL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	vbus_sel_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	vbus_sel_0	0	R/W	Select VBUSEN control 1: VBOUT bit of the VBUS Control register. 0: Port Power bit of the EHCI/OHCI Operational register.

32.3.5 Connection control register (CON_CTRL)

This register controls USB/PHY connection

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	sel_connect	—	—	connect_2	connect_1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	sel_connect	0	R/W	Select USB connection notification control 1: register (controlled by bit0: connect_1 and bit1: connect_2) 0: Control by host controller
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	connect_2	0	R/W	Notice of USB connection (port2) Used to notify UTM + core of USB connection status 0: USB not connected (single end receiver disabled) 1: USB not connected (single end receiver enabled)
0	connect_1	0	R/W	Notice of USB connection (port1) Used to notify UTM + core of USB connection status 0: USB not connected (single end receiver disabled) 1: USB not connected (single end receiver enabled)

32.3.6 Clock status register (CLK_STAT)

This register shows USB/PHY clock status.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	frclk60 lock	frclk48 lock	phy lock	—	—	plllock_ 2	plllock_ 1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	frclk60_lock	0	R	FRCLK60 LOCK status 0: UNLOCK 1: LOCK
5	frclk48_lock	0	R	FRCLK48 LOCK status 0: UNLOCK 1: LOCK
4	phylock	0	R	PHY PLL LOCK Status (Transceiver) 0: UNLOCK 1: LOCK
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	plllock_2	0	R	PLL LOCK status (port2) 0: UNLOCK 1: LOCK
0	plllock_1	0	R	PLL LOCK status (port1) 0: UNLOCK 1: LOCK

32.4 Operation

32.4.1 USB/PHY Initialization

In the initial state, the reset of USB/PHY Port1 and Port2 is controlled by the register in the USBPHY Control Register, and the reset is asserted, so the control is switched to controller by the following initialization.

Also, in the initial state, the USB/PHY PLL Reset is controlled by the Host Controller in the OTG Controller, but it needs to be switched to controlled by the USBPHY Control Register.

- **Initialization when starting only Port1 of USB/PHY (when Port2 is not used)**

  Write H’0000_1133 to the reset register (RESET) of USBPHY Control Register, and then write H’0000_1022 to release the reset.

  (By the above register settings, USBPHY Control Register controls the PLL Reset and the reset is released.

  Port1 of USB/PHY is controlled by Host Controller, and Port2 is reset by USBPHY Control Register.)

- **Initialization when starting only Port2 of USB/PHY (when Port1 is not used)**

  Write H’0000_1133 to the reset register (RESET) of USBPHY Control Register, and then write H’0000_1011 to release the reset.

  (By the above register settings, USBPHY Control Register controls the PLL Reset and the reset is released.

  USB/PHY Port1 is reset by USBPHY Control Register, and port2 is controlled by Host Controller.)

- **Initialization when starting both Port1 and Port2 of USB/PHY**

  Write H’0000_1133 to the reset register (RESET) of USBPHY Control Register, and then write H’0000_1000 to release the reset.

  (By the above register settings, USBPHY Control Register controls the PLL Reset and the reset is released.

  USB/PHY Port1 is controlled by the Host Controller, and Port2 is controlled by the Host Controller.)

The common setting sequence both Host mode and Function mode including USB/PHY initialization is described in **Section 32A.9.1, Host/Peripheral Common Setting Sequence**.

32.4.2 Handling of permanently unused pin

Take the following measures when USB interface is not used permanently.

32.4.2.1 USB/PHY related pins

- When both Port1 and Port2 are unused

Take the following measures for the USB pins and power supply.

- Connect the USB0_DP/DM and USB1_DP/DM to GND via a resistor of 10kΩ.
- Set the RREF to Open.
- Connect USB_VDD18 to a 1.8V power supply. However, there is no need to separate the digital and analog power supplies.
- Connect USB_VDD33 to GND.
- Connect VSS to GND.

Also, perform the following control with software.

- Set the direct power down (via UDIRPD register). See **Section 32.4.3, Direct power down (DIRPD)** for more information.

- When either Port1 or Port2 is unused

Supply all power.

Since USB_VDD18 / USB_VDD33 are common to 2 Port PHY, it is necessary to supply power even when one of the ports is not in use.

Make the following settings for unused port.

- Apply Port reset:
  - Apply reset to unused USB/PHY port by controlling the Reset register (RESET).
  - Set sel_p1reset=1 and phyrst_1=1 when USB/PHY Port1 is unused.
  - Set sel_p2reset=1 and phyrst_2=1 when USB/PHY Port2 is unused.

- Power down setting:
  - Set power down setting to unused USB/PHY port by controlling the Clock control register (UCLKCTL).
  - Set sel_suspendm=1 and suspendm_1=0 when USB/PHY Port1 is unused.
  - Set sel_suspendm=1 and suspendm_2=0 when USB/PHY Port2 is unused.

- Single-ended receiver stop:
  - Stop single-ended receiver of unused USB/PHY port by controlling the Connection control register (CON_CTRL).
  - Set sel_connect=1 and connect_1=0 when USB/PHY Port1 is unused.
  - Set sel_connect=1 and connect_2=0 when USB/PHY Port2 is unused.

32.4.2.2 Handling of USB controller related pins

- When Host controller is unused

The USB0_VBUSEN pin leaves open and fix the USB0_OVRCUR pin to 1 when USB/PHY port1 is unused.

The USB1_VBUSEN pin leaves open and fix the USB1_OVRCUR pin to 1 when USB/PHY port2 is unused.

- Function controller is unused

Fix the USB0_VBUSIN pin to 0 when USB/PHY port1 is unused.

32.4.3 Direct power down (DIRPD)

By directly controlling the power down signal of the USB/PHY with the USBPHY Control register, the USB/PHY can be put into a low power state regardless of the state of the controller.

When set to the direct power down state, all USB/PHY functions including the PLL will be stopped. Direct power down can only be used when the USB/PHY is unused (means both ports are unused).

Follow the procedure below to switch to / return to the direct power down mode.

For details on the USBPHY Control Register reset register and direct power down setting register used in the procedure below, refer to **Section 32.3.1, Reset register (RESET)** and **Section 32.3.3, Direct power down setting register (UDIRPD)**, respectively.

32.4.3.1 Enter to Direct power down mode

Follow the steps below to switch to direct power down mode.

(1) USB operation stop

Stop the operation of USB.

(2) Assert the direct power down signal.

It is asserted by setting H’0000_0010 in the direct power-down setting register (UDIRPD) and then setting H’0000_0011.

There are no timing restrictions when transitioning to direct power down. Immediately after setting, it shifts to power down mode.

32.4.3.2 Resume from Direct power down mode

Follow the steps below to recover from the direct power down mode.

(1) Assert a USB/PHY PLL reset.

The PLL reset is asserted by setting 1 to sel_pllreset bit of the Reset register (RESET) and then setting 1 to pllreset bit.

The reset period should be 1µs or more.

(2) De-assert direct power down signal

Negate the direct power down signal after continuing to assert the PLL reset for more than 1us. It is negated by setting H'0000_0010 in the direct power down setting register (UDIRPD) and then setting H'0000_0000.

After negating the direct power down signal, keep asserting the PLL reset for at least 1µs.

(3) De-assert USB/PHY PLL reset

After negating the direct power down signal, continue to assert the PLL reset for 1us or more, and then negate the PLL reset.

The PLL reset is negated by clearing the pllreset bit in the reset register (RESET) and then clearing the sel_pllreset.

The timing chart when returning from the direct putter down is shown below.

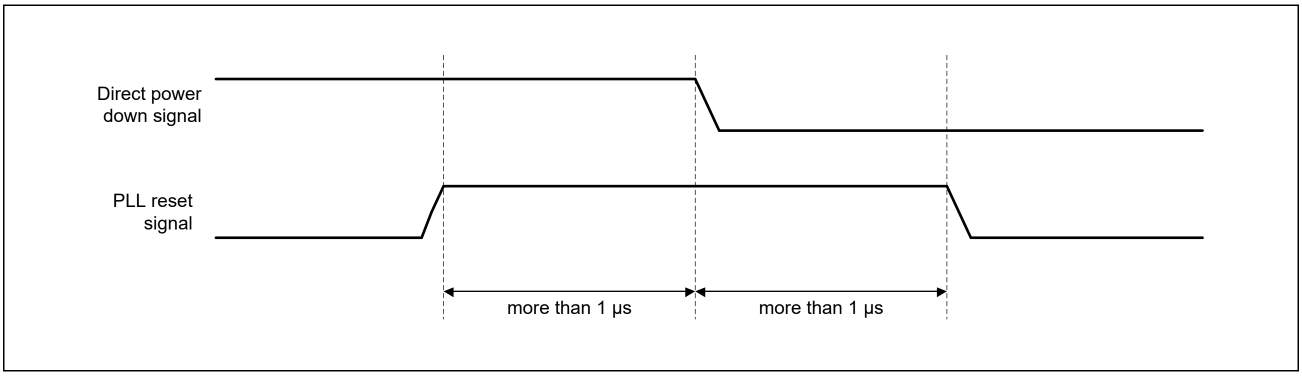


Figure 32.2 Resume from Direct power down mode

32.5 Usage Notes

32.5.1 Precautions in power down mode

When USB interface is unused, it can be set to power-down mode by register setting. For the setting, refer to **Section 32.3.3, Direct power down setting register (UDIRPD)**.

The precautions for power down mode are as follows.

- Once enter to power down, do not access the interface.
- Move to the power-down state after the USB is in unused state. If USB interface moves power-down state while it is operating, current may continue to flow to the USB side and it may damage device.

32A. USB 2.0 Host Module

32A.1 Overview

32A.1.1 Overview

This LSI has two USB 2.0 host/function modules. For each module, you can switch between the host mode and the peripheral mode by specifying the UCOM register setting. This section describes the circuits that are common to both modes, and the host controller itself.

32A.1.2 Features

This module has the following features:

Function	Description
Host function	• Supporting high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) transfers • Compliant with Open Host Controller Interface (OHCI) Specification for USB Revision 1.0a • Compliant with Enhanced Host Controller Interface (EHCI) Specification for USB Revision 1.1*[1]
Other functions	• Battery charging (compliant with Battery Charging Specification Revision 1.2)*[2] • Dual-role-device function (static switching between the USB host and USB peripheral functions)

Note 1. Some functions (specifications) are not supported.

Note 2. The setting for battery charging is handled by the host controller even if the function controller is selected.
This LSI does not support DCP (Dedicated Charging Port) as a Downstream Port.

32A.1.2.1 EHCI v1.1 functions

Conventional host controller modules comply with EHCI Specification Revision.1.0. Meanwhile, this module supports the additional functions listed below to comply with EHCI v1.1 Addendum.

Note, however, that this module supports only the functions marked with a circle “✓” in the table below.

Function Name	Support
Per-Port Change Events	✓
Shorter Periodic Frame List	✓
Hardware Prefetching	—
Link Power Management (LPM)	✓

Of the three features to support, the registers associated with “Per-Port Change Events” and “Shorter Periodic Frame List” are listed below. For more information on each register, refer to the appropriate register specification of the **Section 32A.2, Register Descriptions**.

With regard to “Link Power Management (LPM)”, it is described in **Section 32A.1.2.2, Link Power Management (LPM) function**.

Table 32A.1 Per-Port Change Events

Relevant register	Relevant bit		Attribute	Function
HCCPARAMS (offset: H’108)	bit 18	Per-Port Change Event Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: H’120)	bit 15	Per-Port Change Events Enable	R/W	Per-Port event notification setting. Writing 1b to this bit enables the Per-Port event notification function.
USBSTS (offset: H’124)	bit 16	Port-1 Change Detect	R/W(1)	If a port-1 change event is detected, 1b is written to this bit.
USBINTR (offset: H’128)	bit 16	Port-1 Change Event Enable	R/W	Enable/Disable setting of the above “port-1 Change Detect” field. To reflect the port event on the above field for the port, set corresponding bit to 1b.

Table 32A.2 Shorter Periodic Frame List

Relevant register	Relevant bit		Attribute	Function
HCCPARAMS (offset: H’108)	bit 19	32-Frame Periodic List Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: H’120)	bit[3:2]	Frame List Size	R/W	This bit determines the Frame List Size. To determine 32-frames, set this field 11b.

32A.1.2.2 Link Power Management (LPM) function

This module supports the power management function conforming to “USB 2.0 Link Power Management Addendum to the Universal Serial Bus v2.0 Specification” (abbreviated as LPM).

When the peripheral device which supports LPM function is connected to this module, the device can be moved to suspend state faster than conventional by using LPM function.

*In using LPM function: 8 to 10 µs

The relevant registers are listed in the following. Refer to the appropriate register specification of each register detail in **Section 32A.2, Register Descriptions**.

Relevant register	Relevant bit		Attribute	Function
HCCPARAMS (offset: H’108)	bit 17	Link Power Management Capability	R	This bit indicates to support this function. This bit is fixed to 1b.
USBCMD (offset: H’120)	bit[27:24]	Host-Initiated Resume Duration	R/W	The minimum duration time of K-state drive in resume from LPM state.
PORTSC1 (offset: H’164)	bit[31:25]	Device Address [7:0]	R/W	The device address of the destination of LPM Token. In using LPM function, this setting is needed. Set the address of the device connected to the corresponding port before sending LPM Token.
	bit[24:23]	Suspend Status [1:0]	R	The device response to LPM Token.
	bit 9	Suspend using L1	R/W	In using LPM function, set this bit 1b.
	bit 7	Suspend	R/W	When this bit is set to 1b this module starts LPM Token transaction, in condition of “Suspend using L1 bit = 1b” and “Device address field = H’000”.
PORT_LPM_CTR1 (offset: H’320)	bit[7:4]	NYET_RETRY_CNT_P1[3:0]	R/W	The number of retry for NYET response of device in LPM transaction.
	bit 3	REMOTEWAKE_EN_P1	R/W	Setting of LPM RemoteWakeup permission. 0b: permitted (default) 1b: not permitted
	bit 2	SLEEP_INT_EN_P1	R/W	In LPM transaction, setting to generate an interrupt or not when received a response other than ACK. 0b: not to generate an interrupt (default) 1b: to generate an interrupt
	bit 1	RETRY_ENABLE_NYET_P1	R/W	In LPM transaction, setting to the host behavior when received NYET response from the device. 0b: not to Retry (default) 1b: to Retry
	bit 0	HIRD_SEL_P1	R/W	Setting the duration time of the K-state drive, in resume of LPM state.

32A.1.2.3 OTG function

Detects changing of the USB0_OTG_ID pin by U2H0_OBINT interrupt and switches the Host/Function role. Confirm the IDCHG_STA bit of **Section 32A.2.4.5(2), OTG-BC Interrupt Status Register** and the IDMON bit of **Section 32A.2.4.5(5), Line Control Port 1 Register**.

Host/Function role switching is performed by the OTG_PERI bit of **Section 32A.2.4.5(1), Common Control Register**.

OTG_PERI bit
0 : Host mode
1 : Function mode (Only CH0 can be set)

32A.1.2.4 Battery-charging function

The Control and monitoring of the Battery Charging I/F of the USBPHY is set in the UCOM register of this module. The detail of the UCOM register, refer to **Section 32A.2.4.5, UCOM Register**.

32A.1.2.5 Suspend extension function

This module implements the following two suspend extension functions to reduce power consumption by stopping PLL of USBPHY.

The detail of the register, refer to **Section 32A.2.4.4(6), Suspend Control Register**.

[Relevant register]

Suspend Control Register (offset: H’308)

[Functional specification]

Function		Relevant bit	
[1]	Function to assert USBPHY SUSPENDM by asserting the Suspend bit in the OHCI/EHCI Operational Register	bit 31	SUSPENDM_ENABLE
[2]	Function to forcibly assert USBPHY SUSPENDM	bit 0	GLOBAL_SUSPENDM_P1

32A.1.3 Support of USB-Related Specifications

USB-Related Specification or Function			Support
Host function	High Speed	Bulk IN/OUT transfer	✓
		Control IN/OUT transfer	✓
		Isochronous IN/OUT transfer	✓
		Isochronous Highband transfer	✓
		Interrupt IN/OUT transfer	✓
	Full Speed	Bulk IN/OUT transfer	✓
		Control IN/OUT transfer	✓
		Isochronous IN/OUT transfer	✓
		Interrupt IN/OUT transfer	✓
	Low Speed	Control IN/OUT transfer	✓
		Interrupt IN/OUTtransfer	✓
	No. of hub connection stages	HS: 5 stages	✓
		FS: 5 stages	✓
	Support of EHCI V1.1	Hardware Prefetching	×
		Link Power Management	✓
		Per-Port Change Events	✓
		Shorter Periodic Frame List	✓
Battery-charging function (hereafter called the “BC function”)			✓
Dual role device	* Function to statically switch between the host and peripheral modes		✓

Remarks: ✓: support
×: no support

32A.2 Register Descriptions

32A.2.1 Register Attributes

Table 32A.3 Register Attributes

Register Attribute	Description
R/W	Register bits can be read and written.
R/W(1)	Register bits can be read. A clear bit may be set by writing “1”; writing 0 to R/W(1) bits has no effect.
R/W(0)	Register bits can be read. A clear bit may be set by writing “0”; writing 1 to R/W(0) bits has no effect.
R	Register bits can only be read.
W	Register bits can only be written.
Reserved	Reserved bits are Read Only field.

32A.2.2 Base Address

Table 32A.4 Base Addresses for Each Channel of the USB Host Module

Channel	Base Address	
0	H’0_11C5_0000	(Cortex-A55 Address Space)
	H’41C5_0000	(Cortex-M33 Address Space Non-Secure)
	H’51C5_0000	(Cortex-M33 Address Space Secure)
1	H’0_11C7_0000	(Cortex-A55 Address Space)
	H’41C7_0000	(Cortex-M33 Address Space Non-Secure)
	H’51C7_0000	(Cortex-M33 Address Space Secure)

32A.2.3 Register Overview

Register Name	Abbreviated Name	R/W	Initial Value	Offset Address	Access Size	Remarks
HcRevision	HcRevision	R	H'0000_0000	H'000	32	OHCI Operation Registers
HcControl	HcControl	R/W	H'0000_0000	H'004	32	
HcCommandStatus	HcCommandStatus	R/W	H'0000_0000	H'008	32	
HcInterruptStatus	HcInterruptStatus	R/W	H'0000_0000	H'00C	32	
HcInterruptEnable	HcInterruptEnable	R/W	H'0000_0000	H'010	32	
HcInterruptDisable	HcInterruptDisable	R/W	H'0000_0000	H'014	32	
HcHCCA	HcHCCA	R/W	H'0000_0000	H'018	32	
HcPeriodCurrentED	HcPeriodCurrentED	R	H'0000_0000	H'01C	32	
HcControlHeadED	HcControlHeadED	R/W	H'0000_0000	H'020	32	
HcControlCurrentED	HcControlCurrentED	R/W	H'0000_0000	H'024	32	
HcBulkHeadED	HcBulkHeadED	R/W	H'0000_0000	H'028	32	
HcBulkCurrentED	HcBulkCurrentED	R/W	H'0000_0000	H'02C	32	
HcDoneHead	HcDoneHead	R	H'0000_0000	H'030	32	
HcFmInterval	HcFmInterval	R/W	H'0000_2EDF	H'034	32	
HcFmRemaining	HcFmRemaining	R	H'0000_2EDF	H'038	32	
HcFmNumber	HcFmNumber	R	H'0000_0000	H'03C	32	
HcPeriodicStart	HcPeriodicStart	R/W	H'0000_0000	H'040	32	
HcLSThreshold	HcLSThreshold	R/W	H'0000_0628	H'044	32	
HcRhDescriptorA	HcRhDescriptorA	R/W	H'0F00_0901	H'048	32	
HcRhDescriptorB	HcRhDescriptorB	R/W	H'0002_0000	H'04C	32	
HcRhStatus	HcRhStatus	R/W	H'0000_0000	H'050	32	
HcRhPortStatus1	HcRhPortStatus1	R/W	H'0000_0000	H'054	32	
HCIVERSION / CAPLENGTH	CAPL_VERSION	R	H'0110_0020	H'100	32	EHCI Capability Registers
HCSPARAMS	HCSPARAMS	R	H'0000_1191	H'104	32	
HCCPARAMS	HCCPARAMS	R	H'000E_0006	H'108	32	
HCSP_PORTROUTE	HCSP_PORTROUTE	R	H'0000_0000	H'10C	32	
USBCMD	USBCMD	R/W	H'0008_0B00	H'120	32	EHCI Operation Registers
USBSTS	USBSTS	R/W	H'0000_1000	H'124	32	
USBINTR	USBINTR	R/W	H'0000_0000	H'128	32	
FRINDEX	FRINDEX	R/W	H'0000_0000	H'12C	32	
CTRLDSSEGMENT	CTRLDSSEGMENT	R	H'0000_0000	H'130	32	
PERIODICLISTBASE	PERIODICLISTBASE	R/W	H'0000_0000	H'134	32	
ASYNCLISTADDR	ASYNCLISTADDR	R/W	H'0000_0000	H'138	32	
CONFIGFLAG	CONFIGFLAG	R/W	H'0000_0000	H'160	32	
PORTSC1	PORTSC1	R/W	H'0000_2000	H'164	32	
INT_ENABLE	INT_ENABLE	R/W	H'0000_0000	H'200	32	AHB Registers
INT_STATUS	INT_STATUS	R/W	H'0000_0000	H'204	32	
AHB_BUS_CTR	AHB_BUS_CTR	R/W	H'0000_0000	H'208	32	
USBCTR	USBCTR	R/W	H'0000_0002	H'20C	32	

Register Name	Abbreviated Name	R/W	Initial Value	Offset Address	Access Size	Remarks
Register Enable/Clock Gating Control	REGEN_CG_CTRL	R/W	H'0000_0000	H'304	32	Core Defined Registers
Suspend Control	SPD_CTRL	R/W	H'0000_0000	H'308	32	
Suspend/Resume Timer Setting	SPD_RSM_TIMSET	R/W	H'01F4_03E8	H'30C	32	
Overcurrent Detection/Sleep Timer Setting	OC_SLP_TIMSET	R/W	H'0C83_0D40	H'310	32	
SBRN/FLADJ/PORTWAKECAP	SBRN_FLADJ_PW	R/W	H'0003_2020	H'314	32	
PORT_LPM_CTRL1	PORT_LPM_CTRL1	R/W	H'0000_0000	H'320	32	
Common Control	COMMCTRL	R/W	H'8000_0000	H'800	32	OTG/BC Module Control Register
OTG-BC Interrupt Status	OBINTSTA	R/W	H'0000_0001	H'804	32	OTG/BC Interrupt Status Register
OTG-BC Interrupt Enable	OBINTEN	R/W	H'0000_0000	H'808	32	OTG/BC Interrupt Enable Register
VBUS Control	VBCTRL	R/W	H'0001_0000	H'80C	32	OTG VBUS Control Register
Line Control Port 1	LINECTRL1	R/W	H'0000_0000	H'810	32	OTG USB Bus Control Register (Port 1)
BC Control Port 1	BCCTRL1	R/W	H'0300_0000	H'820	32	Battery Charging Control Register (Port 1)

32A.2.4 Description of Registers

32A.2.4.1 OHCI Operational Register

(1) HcRevision Register

Abbreviated name of register: HcRevision

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	Revision							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should always be 0.
7 to 0	Revision	All 0	R	This field indicates the version of HCI specifications implemented in this host controller module. Because this module conforms to OHCI standard 1.0a, H'10 is indicated.

(2) HcControl Register

Abbreviated name of register: HcControl

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RWC	—	HCFS		BLE	CLE	IE	PLE	CBSR	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved The write value should always be 0.
9	RWC	0	R/W	Remote Wakeup Connected This bit indicates whether the host controller supports remote wakeup signaling. Software should set this bit to 1b in the initialization sequence if it is required to support remote wakeup. Note that this bit can be initialized at hardware reset only. 0b: The remote wakeup is not supported. 1b: The remote wakeup is supported
8	—	0	R	Reserved The write value should always be 0.
7, 6	HCFS	All 0	R/W	Host Controller Functional State This field indicates the operating state of the host controller. 00b: USB Reset 01b: USB Resume 10b: USB Operational 11b: USB Suspend When the state is changed to USB Operational, the host controller module starts SOF transmission at 1 ms boundary. This field is basically controlled by software, but it can be controlled by the host controller in the state of USB Suspend only. If the host controller (in the state of USB Suspend) detects a Remote Wakeup signal from the connecting device, the state in this field is changed to USB Resume. Note that the reset value of this field differs between hardware reset and software reset. Hardware reset: 00b (USB Reset) Software reset: 11b (USB Suspend)
5	BLE	0	R/W	BulkList Enable This bit sets whether the bulk list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the bulk list, this bit must be 0b. 0b: The processing of the Bulk list is disabled. 1b: The processing of the Bulk list is enabled
4	CLE	0	R/W	ControlList Enable This bit sets whether the control list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the control list, this bit must be 0b. 0b: The processing of the Control list is disabled. 1b: The processing of the Control list is enabled.

Bit	Bit Name	Initial Value	R/W	Description
3	IE	0	R/W	Isochronous Enable This bit sets whether the isochronous ED processing is performed. The setting value of this bit is enabled from the next frame. If the host controller module detects isochronous ED (F = 1) during the periodic list processing, it checks the bit and determines whether to perform isochronous ED processing. 1b: The processing of the isochronous ED is continued. 0b: The periodic list processing is stopped, and the bulk/control list processing is started. 0b: The processing of the isochronous ED is disabled. 1b: The processing of the isochronous ED is enabled.
2	PLE	0	R/W	Periodic List Enable This bit indicates whether the periodic list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. The host controller module checks this bit before starting the periodic list processing. 0b: The processing of the periodic list is disabled. 1b: The processing of the periodic list is enabled.
1, 0	CBSR	All 0	R/W	Control Bulk Service Ratio This field defines the service ratio of the control transfer and bulk transfer. When the periodic list is processed, the service ratio defined in this field is used for transfer.

CBSR	No. of Control EDs Over Bulk EDs Served
00b	1 : 1
01b	2 : 1
10b	3 : 1
11b	4 : 1

(3) HcCommandStatus Register

Abbreviated name of register: HcCommandStatus

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOC	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	BLF	CLF	HCR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17, 16	SOC	All 0	R	Scheduling Overrun Count This field counts the number of scheduling overrun. This field is initialized to 00b, and is counted up every time scheduling overrun is detected. After the field is incremented to 11b, it returns to 00b. Even if the SO (Scheduling Overrun) bit in the HcInterrupt Status register is set, this field is counted up when scheduling overrun is detected.
15 to 3	—	All 0	R	Reserved The write value should always be 0.
2	BLF	0	R/W	BulkList Filled This bit indicates whether TD exists in the bulk list. To add TD to ED in the bulk list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the bulk list. If this bit is set to 0b, the host controller does not start the list processing. If this bit is set to 1b, the host controller starts processing of the bulk list, and sets this bit to 0b. When the host controller finds TD in the bulk list, it sets this bit to 1b again, and continues processing of the bulk list. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the bulk list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the BLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the bulk list. 1b: TD exists in the bulk list.

Bit	Bit Name	Initial Value	R/W	Description
1	CLF	0	R/W	ControlList Filled This bit indicates whether TD exists in the control list. To add TD to ED in the control list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the control list. If this bit is set to 0b, the host controller does not start the processing of the control list. If this bit is set to 1b, the host controller starts processing of the control list, and set this bit to 0b. When the host controller finds TD in the control list, it sets this be to 1b again, and continues the list processing. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the control list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the CLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the control list. 1b: TD exists in the control list.
0	HCR	0	W	Host Controller Reset This bit is used to start OHCI software reset for the host controller. When this bit is set to 1b, the operating status of the host controller is changed to USB Suspend regardless of the functional state of the host controller. Also, the most OHCI Operational registers and OHCI control circuits are initialized. When the software reset finishes, the host controller clears this bit to 0b.

(4) HcInterruptStatus Register

Abbreviated name of register: HcInterruptStatus

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSC	FNO	UE	RD	SF	WDH	SO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved The write value should always be 0.
6	RHSC	0	R/W	Root Hub Status Change Interrupt bit that indicates that the state of the HcRhStatus register or HcRhPortStatus1 register has changed. When the Root Hub status is changed, the host controller sets this bit to 1b. Writing 1b to this bit clears the interrupt. 0b: The status of root hub has not changed. 1b: The status of root hub has changed.
5	FNO	0	R/W	Frame Number Overflow Interrupt bit that indicates that the MSB in the FrameNumber (FN) field of the HcFmNumber register has changed. When the MSB in the Frame Number field is changed from 0 to 1, or from 1 to 0, this bit is set after HccaFrameNumber is updated. Writing this bit to 1b clears the interrupt. 0b: The overflow of frame number has not occurred. 1b: The overflow of frame number has occurred.
4	UE	0	R/W	Unrecoverable Error Interrupt bit that indicates that a system error that is not related to the USB (for example, an error on the system bus) has been detected. Writing this bit to 1b clears the interrupt. 0b: The unrecoverable error has not occurred. 1b: The unrecoverable error has occurred.
3	RD	0	R/W	Resume Detected Interrupt bit that indicates that Resume has been detected. When the host controller detects the Resume signal (RemoteWakeup) from an USB device, it sets this bit to 1b. This bit is not set when the Resume signal is sent by setting the HCFS field to USB Resume. Writing this bit to 1b clears the interrupt. 0b: The host controller has not detected resume signaling (RemoteWakeup). 1b: The host controller has detected resume signaling (RemoteWakeup).

Bit	Bit Name	Initial Value	R/W	Description
2	SF	0	R/W	Start of Frame Interrupt bit that indicates that Hcca Frame Number was updated when each frame started. The host controller sends an SOF packet and updates HccaFrameNumber at the same time, and sets this bit to 1b. Writing this bit to 1b clears the interrupt. 0b: The host controller has not started new frame. 1b: The host controller has started new frame.
1	WDH	0	R/W	Writeback DoneHead Interrupt bit that indicates that the host controller has written the contents of HcDoneHead to HccDoneHead. The host controller sets this bit to 1b immediately after it updates HccaDoneHead. Note that HccaDoneHead is not updated until this bit is cleared. Writing this bit to 1b clears the interrupt. This bit must be cleared only after the contents of HccaDoneHead are saved. 0b: The write back to HccaDoneHead has not occurred. 1b: The write back to HccaDoneHead has occurred.
0	SO	0	R/W	Scheduling Overrun Interrupt bit that indicates that overrun of the USB schedule occurred. When USB scheduling overrun occurs, the host controller updates HccaFrameNumberl, and sets this bit to 1b. When this bit is set, the SchedulingOverrunCount field of the HcCommandStatus register is also incremented. Writing this bit to 1b clears the interrupt. 0b: The scheduling overrun has not occurred. 1b: The scheduling overrun has occurred.

(5) HcInterruptEnable Register

Abbreviated name of register: HcInterruptEnable

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MIE	OCE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSCE	FNOE	UEE	RDE	SFE	WDHE	SOE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MIE	0	R/W	Master Interrupt Enable This bit sets whether each Interrupt Enable setting (which is set in HcInterruptEnable [30:0]) is enabled. If this bit is set to 0b, all OHCI interrupts are masked. To clear this bit, set the MID bit (bit 31) of the HcInterruptDisable register to 1b. 0b: All interrupts are disabled. 1b: Interrupts that are set to 1b are enabled.
30	OCE	0	R/W	OC (Ownership Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes OC (OwnershipChange). To clear this bit, set the OCD bit (bit 30) of the HcInterruptDisable register to 1b. 0b: OC (OwnershipChange) interrupt is disabled. 1b: OC (OwnershipChange) interrupt is enabled.
29 to 7	—	All 0	R	Reserved The write value should always be 0.
6	RHSCE	0	R/W	RHSC (RootHub Status Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RHSC (RootHub Status Change). To clear this bit, set the RHSCD bit (bit 6) of the HcInterruptDisable register to 1b. 0b: RHSC (RootHub Status Change) interrupt is disabled. 1b: RHSC (RootHub Status Change) interrupt is enabled.
5	FNOE	0	R/W	FNO (Frame Number Overflow) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes FNO (Frame Number Overflow). To clear this bit, set the FNOD bit (bit 5) of the HcInterruptDisable register to 1b. 0b: FNO (Frame Number Overflow) interrupt is disabled. 1b: FNO (Frame Number Overflow) interrupt is enabled.
4	UEE	0	R/W	UE (Unrecoverable Error) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes UE (Unrecoverable Error). To clear this bit, set the UED bit (bit 4) of the HcInterruptDisable register to 1b. 0b: UE (Unrecoverable Error) interrupt is disabled. 1b: UE (Unrecoverable Error) interrupt is enabled.
3	RDE	0	R/W	RD (Resume Detect) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RD (Resume Detect). To clear this bit, set the RDD bit (bit 3) of the HcInterruptDisable register to 1b. 0b: RD (Resume Detect) interrupt is disabled. 1b: RD (Resume Detect) interrupt is enabled.

Bit	Bit Name	Initial Value	R/W	Description
2	SFE	0	R/W	SF (Start of Frame) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SF (Start of Frame). To clear this bit, set the SFD bit (bit 2) of the HcInterruptDisable register to 1b. 0b: SF (Start of Frame) interrupt is disabled. 1b: SF (Start of Frame) interrupt is enabled.
1	WDHE	0	R/W	WDH (Writeback DoneHead) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes WDH (Writeback DoneHead). To clear this bit, set the WDHDD bit (bit 1) of the HcInterruptDisable register to 1b. 0b: WDH (Writeback DoneHead) interrupt is disabled. 1b: WDH (Writeback DoneHead) interrupt is enabled.
0	SOE	0	R/W	SO (Scheduling Overrun) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SO (Scheduling Overrun). To clear this bit, set the SOD bit (bit 0) of the HcInterruptDisable register to 1b. 0b: SO (Scheduling Overrun) interrupt is disabled. 1b: SO (Scheduling Overrun) interrupt is enabled.

(6) HcInterruptDisable Register

Abbreviated name of register: HcInterruptDisable

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MID	OCD	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W(1)	R/W(1)	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSCD	FNOD	UED	RDD	SFD	WDHD	SOD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)

Bit	Bit Name	Initial Value	R/W	Description
31	MID	0	R/W (1)	Master Interrupt Disable This bit sets whether the Enable setting (for each interrupt) that is set by HcInterruptEnable[30:0] is disabled. If this bit is set to 1b, the MIE bit (bit 31) of the HcInterruptEnable register is cleared to 0b, and all OHCI interrupts are masked. Writing 0b to this bit is ignored.
30	OCD	0	R/W (1)	OC (Ownership Change) Interrupt Disable bit. If this bit is set to 1b, the OCE bit (bit 30) of the HcInterruptEnable register is cleared to 0b, and OC (Ownership Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
29 to 7	—	All 0	R	Reserved The write value should always be 0.
6	RHSCD	0	R/W (1)	RHSC (RootHub Status Change) Interrupt Disable bit. If this bit is set to 1b, the RHSCE bit (bit 6) of the HcInterruptEnable register is cleared to 0b, and RHSC (RootHub Status Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
5	FNOD	0	R/W (1)	FNO (Frame Number Overflow) Interrupt Disable bit. If this bit is set to 1b, the FNOE bit (bit 5) of the HcInterruptEnable register is cleared to 0b, and FNO (Frame Number Overflow) is excluded from the interrupt source. Writing 0b to this bit is ignored.
4	UED	0	R/W (1)	UE (Unrecoverable Error) Interrupt Disable bit. If this bit is set to 1b, the UEE bit (bit 4) of the HcInterruptEnable register is cleared to 0b, and UE (Unrecoverable Error) is excluded from the interrupt source. Writing 0b to this bit is ignored.
3	RDD	0	R/W (1)	RD (Resume Detected) Interrupt Disable bit. If this bit is set to 1b, the RDE bit (bit 3) of the HcInterruptEnable register is cleared to 0b, and RD (Resume Detected) is excluded from the interrupt source. Writing 0b to this bit is ignored.
2	SFD	0	R/W (1)	SF (Start of Frame) Interrupt Disable bit. If this bit is set to 1b, the SFE bit (bit 2) of the HcInterruptEnable register is cleared to 0b, and SF (Start of Frame) is excluded from the interrupt source. Writing 0b to this bit is ignored.
1	WDHD	0	R/W (1)	WDH (Writeback DoneHead) Interrupt Disable bit. If this bit is set to 1b, the WDHE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and WDH (Writeback DoneHead) is excluded from the interrupt source. Writing 0b to this bit is ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	SOD	0	R/W (1)	SO (Scheduling Overrun) Interrupt Disable bit. If this bit is set to 1b, the SOE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and SO (Scheduling Overrun) is excluded from the interrupt source. Writing 0b to this bit is ignored.

(7) HcHCCA Register

Abbreviated name of register: HcHCCA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HcHCCA[31:24]								HcHCCA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HcHCCA[15:8]								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	HcHCCA	All 0	R/W	This field sets the base address (of the RAM) that is assigned as the Host Controller Communication Area. This field must be set at initialization. The host controller requests (as HCCA) 256-byte area from the base address specified in this field.
7 to 0	—	All 0	R	Reserved The write value should always be 0.

(8) HcPeriodicCurrentED Register

Abbreviated name of register: HcPeriodCurrentED

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PECD[31:24]								PECD[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PECD[15:8]								PECD[7:4]				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	PECD	All 0	R	Period Current ED This pointer indicates the physical address of ED in the periodic list that is currently processed. The host controller updates this pointer when the list processing of the periodic ED finishes.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(9) HcControlHeadED Register

Abbreviated name of register: HcControlHeadED

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CHED[31:24]								CHED[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CHED[15:8]								CHED[7:4]				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CHED	All 0	R/W	Control Head ED This field specifies the physical address of the head ED of the control list. These bits must be set for control transfer before the CLE bit of the HcControl register is set. The host controller starts processing of the control list from the HcBulkHeadED pointer.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(10) HcControlCurrentED Register

Abbreviated name of register: HcControlCurrentED

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CCED[31:24]								CCED[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCED[15:8]								CCED[7:4]				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CCED	All 0	R/W	Control Current ED This pointer indicates the physical address of the ED that is currently processed in the control list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the control list is reached, the host controller checks the ControlListFilled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcControlHeadED field are copied to the HcControlCurrentED field, and the ControlListFilled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the ControlListEnable bit of the HcControl register is cleared. If the ControlListEnable bit is set to 1b, the value of this register is only read. This register is initially set to H'0 to indicate the end of the control list.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(11) HcBulkHeadED Register

Abbreviated name of register: HcBulkHeadED

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BHED[31:24]								BHED[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BHED[15:8]								BHED[7:4]				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BHED	All 0	R/W	BulkHead ED This field specifies the physical address of the head ED of the bulk list. This field must be set for bulk transfer before the BLE bit of the HcControl register is set. The host controller starts the processing of the control list from the HcBulkHeadED pointer.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(12) HcBulkCurrentED Register

Abbreviated name of register: HcBulkCurrentED

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BCED[31:24]								BCED[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BCED[15:8]								BCED[7:4]				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BCED	All 0	R/W	Bulk Current ED This pointer indicates the physical address of the ED that is currently processed in the bulk list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the bulk list is reached, the host controller checks the ControlListFilled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcBulkHeadED field are copied to the HcBulklCurrentED field, and the ControlListFilled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the ControlListEnable bit of the HcControl register is cleared. If the ControlListEnable bit is set to 1b, the value of this register is only read. This register is initially set to H’0 to indicate the end of the bulk list.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(13) HcDoneHead Register

Abbreviated name of register: HcDoneHead

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DH[31:24]								DH[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DH[15:8]								DH[7:4]				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	DH	All 0	R	Done Head This field indicates the physical address of the HcDoneHead of the host controller. The physical address of the TD that lately finished and is added to the Done queue. After the TD processing finishes, the host controller writes the contents of the HcDoneHead to the NextTD field of the TD. At the same time, the host controller overwrites the contents of HcDoneHead with the TD address. After the host controller writes the contents of this register into HCCA, it sets 0b to this register. Then, the WritebackDoneHead bit of the HcInterruptStatus register is set to 1b.
3 to 0	—	All 0	R	Reserved The write value should always be 0.

(14) HcFmInterval Register

Register abbreviation: HcFmInterval

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIT	FSMPS[30:24]							FSMPS[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	FI[13:8]						FI[7:0]							
Initial Value	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIT	0	R/W	Frame Interval Toggle This bit is used to synchronize the frame setting value with the hardware (host controller). When the FI field is updated, the toggle value is written to this bit. When the FI field value is applied to the FR field of the HcFmRemaining register, the value of this bit is also applied to the FRT bit of the HcFmRemaining register. By checking the toggle value in the FRT bit of the cFmRemaining register, the software can check whether the value in the FI field has been applied to the FR field of the HcFmRemaining register.
30 to 16	FSMPS	All 0	R/W	FS Largest Data Packet This field sets the maximum amount of data (bits) that the host controller can send and receive without the fear of schedule overrun. The host controller compares the current frame position and the setting value in this field to determine the length (of the frame) that is ready to be transferred. This value differs depending on the capacity of the system bus and other reasons, estimate the value and set it to this field. *Note*: The maximum setting value for this field is H’2778. Do not set any value that is larger than H’2778.
15, 14	—	All 0	R	Reserved The write value should always be 0.
13 to 0	FI	H’2EDF	R/W	Frame Interval This field is used to set the length of the frame (bit time) used for Full Speed. Set the value of this field to “H’2EDF” so that 1 frame (= 1 ms) of USB standard is satisfied.

(15) HcFmRemaining Register

Register abbreviation: HcFmRemaining

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FRT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	FR[13:8]						FR[7:0]							
Initial Value	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	FRT	0	R	Frame Remaining Toggle This bit is used to synchronize the frame setting value with the hardware (host controller). When the FR (Frame Remaining) field is set to H’0000, the host controller copies the FI (Frame Interval) field value to the FR field, and copies the FIT (Frame Interval Toggle) bit value to this bit. This bit can be used to check that the FI field of the HcFmInterval register has been correctly copied to the FR field.
30 to 14	—	All 0	R	Reserved The write value should always be 0.
13 to 0	FR	H’2EDF	R	Frame Remaining This field indicates the current frame value for 14-bit down counter. The value in this field counts down as time passes. When the value becomes H’0000, the value of FI (Frame Interval) of the HcFmInterval register is loaded. When the state of the host controller is changed to the USB Operational state, the host controller reloads the value in the FI (Frame Interval) field of the HcFmInterval register, and the new value is used from the next SOF.

(16) HcFmNumber Register

Register abbreviation: HcFmNumber

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FN[15:8]								FN[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved The write value should always be 0.
15 to 0	FN	All 0	R	Frame Number This field indicates the number of passed frames. When the HcFmRemaining register is reloaded, this field is incremented. If this field is reached to H’FFFF, the value is rolled over to H’0000. When the state of the host controller is changed to the USB Operational state, this field is automatically incremented. After the host controller increments the frame number at the frame boundary and sends SOF, the contents of this field are written to HCCA. This is performed before the host controller reads the first ED of the frame. After writing to HCCA, the host controller sets the SF bit of the HcInterruptStatus register.

(17) HcPeriodicStart Register

Register abbreviation: HcPeriodicStart

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PS[13:8]						PS[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved The write value should always be 0.
13 to 0	PS	All 0	R/W	Periodic Start This field indicates the time the host controller starts the periodic list processing in the frame. Estimate an appropriate value, and set the value to this field at the initial setting of the host controller. OHCI standard recommends that you set this setting value to about 90% of the FI field value of the HcFmInterval register. The recommended value is H’2A2F. When the value in the FR field of the HcFmRemaining register reaches the value set to this field, the periodic list processing is given priority over the control/bulk list processing. Therefore, after the currently running control or bulk transfer finishes, the host controller starts the Interrupt list processing.

(18) HcLSThreshold Register

Register abbreviation: HcLSThreshold

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LST[11:8]				LST[7:0]							
Initial Value	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved The write value should always be 0.
11 to 0	LST	H'628	R/W	LS Threshold This field indicates the threshold value of whether transfer is available for the remaining time of the LS transfer frame. If the value of the FR field of the FmRemaining register is larger than the value set to this field, the host controller can start LS transfer.

(19) HcRhDescriptorA Register

Register abbreviation: HcRhDescriptorA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POTPGT								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	NOCP	OCPM	DT	NPS	PSM	NDP							
Initial Value	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	POTPGT	H’0F	R/W	PowerOn To PowerGood Time This field indicates the time for waiting until software can access the root hub port after the power is supplied to the port. The unit of time is 2 ms. Therefore, the wait time is POTPGT × 2 ms.
23 to 13	—	All 0	R	Reserved The write value should always be 0.
12	NOCP	0	R/W	No OverCurrent Protection This bit sets whether the overcurrent function of the root hub is supported. If this bit is set to 0b, the overcurrent state is reported depending on the setting of the OCPM bit. 0b: The overcurrent state is supported. 1b: The overcurrent state is not supported. If you do not require the overcurrent function, set this bit to 1b once the module is released from module standby. For details, see **Section 32A.9.1, Host/Peripheral Common Setting Sequence**.
11	OCPM	1	R/W	OverCurrent Protection Mode This bit sets how to report the overcurrent state of the root hub. If this bit is reset, this bit must indicate the same mode as the PSM (Power Switching Mode) bit. This bit is valid only when the NOCP (NoOverCurrent Protection) bit is cleared (0b). 0b: The overcurrent state is collectively reported for all ports. 1b: The overcurrent state is reported for each port.
10	DT	0	R	Device Type This bit indicates that the root hub is not a composite device. This bit is always 0b because the root hub is not allowed to be a composite device.
9	NPS	0	R/W	No Power Switching This bit sets how to control the port power. If this bit is set to 0b, the PSM bit is used to set whether the power control is collectively performed for all ports or is performed for each port. 0b: The port power can be switched between on and off. 1b: The power is always on while the host controller is running.
8	PSM	1	R/W	Power Switching Mode This bit sets how to control the power switch for each port of the root hub. This bit is valid only when the NPS bit is 0b. 0b: The power of all ports is collectively controlled. 1b: The power of ports is controlled for each port. If the PPCM (PortPower Control Mask) bit of the HcRhDescriptorB register is set, each port responds only to the Set/ClearPortPower command. If the PPCM bit is cleared, each port is controlled by the Set/ClearGlobalPower command.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	NDP	H’01	R	Number Downstream Ports This field indicates the number of downstream ports supported by the root hub.

(20) HcRhDescriptorB Register

Register abbreviation: HcRhDescriptorB

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PPCM	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DR	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17	PPCM	1	R/W	PortPower Control Mask This bit sets the port power control command when the PSM (Power Switching Mode) bit of the HcRhDescriptorA register is set. If this bit is 0b, the Global Power Control command (Set/ClearGlobalPower) is used for control. If this bit is 1b, the Port Power Control command (Set/ClearPortPower) is used for control. If the PSM bit is 0b, this bit is ignored.
16 to 2	—	All 0	R	Reserved The write value should always be 0.
1	DR	0	R/W	Device Removable This bit indicates whether each port of the root hub is removable. If this bit is 0b, the connected device is removable. If this bit is 1b, the connected device is not removable.
0	—	0	R	Reserved The write value should always be 0.

(21) HcRhStatus Register

Register abbreviation: HcRhStatus

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCIC	LPSC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCI	LPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31</td><td>CRWE</td><td>0</td><td>W</td><td>Clear RemoteWakeup Enable
This bit is used to clear the DRWE bit.
If this bit is set to 1b, DRWE (Device RemoteWakeup Enable) bit can be cleared.
Writing 0b to this bit has no effect.</td></tr>
<tr><td>30 to 18</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>17</td><td>OCIC</td><td>0</td><td>R/W (1)</td><td>OverCurrent Indicator Change
This bit is used to report the change in the OCI bit.
If there is any change in the OCI bit, the host controller sets this bit to 1b.
If 1b is written to this bit while this bit is set to 1b, this bit can be cleared.
Writing 0b to this bit has no effect.
0b: There is no change in the Overcurrent state.
1b: There is a change in the Overcurrent state.</td></tr>
<tr><td>16</td><td>LPSC</td><td>0</td><td>R/W</td><td>The meaning of this bit differs depending on whether the operation is read or write.
[Read] Local Power Status Change
This bit is always read as 0b because the Local Power Status is not supported.
[Write] Set Global Power
If this bit is set to 1b, the power to the ports is turned on.
The ports whose power is turned on are determined by the settings of the PSM (Power Switching Mode) bit and the PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register.
<table>
<tr><th>written to this bit</th><th>PSM</th><th>PPCM[N]</th><th>Description</th></tr>
<tr><td>0</td><td>—</td><td>—</td><td>Setting ignored</td></tr>
<tr><td>1</td><td>0</td><td>—</td><td>1b is set to the PPS bit.</td></tr>
<tr><td></td><td>1</td><td>0</td><td>1b is set to the PPS bit.</td></tr>
<tr><td></td><td></td><td>1</td><td>Setting ignored</td></tr>
</table>
Writing 0b to this bit has no effect.</td></tr>
<tr><td>15</td><td>DRWE</td><td>0</td><td>R/W</td><td>[Read] Device RemoteWakeup Enable
This bit sets whether the RemoteWakeup event includes the CSC (Connect Status Change) bit.
If this bit is set to 1b, the CSC bit of the HcRhPortStatus register becomes valid as the Resume event. If the CSC bit is changed to 1b, the state is changed from USB Suspend to USB Resume, and the Resume detection interrupt occurs.
0b: Connect Status Change is not the source of RemoteWakeup.
1b: Connect Status Change is the source of RemoteWakeup.
[Write] Set RemoteWakeup Enable
This bit is used to set the DRWE bit.
Writing 0b to this bit has no effect.</td></tr>
</table>

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>14 to 2</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>1</td><td>OCI</td><td>0</td><td>R</td><td>Over Current Indicator
This bit is used to report the overcurrent state in the global overcurrent detection mode (OCPM bit = 0b).
This bit always indicates 0b when overcurrent for each port is reported (when OPCM bit = 1b).
0b: The port state is normal.
1b: The port is in the overcurrent state.</td></tr>
<tr><td>0</td><td>LPS</td><td>0</td><td>R/W</td><td>The meaning of this bit differs depending on whether the operation is read or write.
[Read] Local Power Status
This bit is always read as 0b because the Local Power Status is not supported.
[Write] Clear Global Power
If this bit is set to 1b, the power to the ports is turned off.
The ports whose power is turned off are determined by the settings of the PSM (Power Switching Mode) bit and PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register.
<table>
<tr><th>written to this bit</th><th>PSM</th><th>PPCM[N]</th><th>Description</th></tr>
<tr><td>0</td><td>—</td><td>—</td><td>Setting ignored.</td></tr>
<tr><td>1</td><td>0</td><td>—</td><td>The PPS bit is cleared to 0b.</td></tr>
<tr><td></td><td>1</td><td>0</td><td>The PPS bit is cleared to 0b.</td></tr>
<tr><td></td><td></td><td>1</td><td>Setting ignored.</td></tr>
</table>
Writing 0b to this bit has no effect.</td></tr>
</table>

(22) HcRhPortStatus[1:NDP] Register

Register abbreviation: HcRhPortStatus1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PRSC	OCIC	PSSC	PESC	CSC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	LSDA	PPS	—	—	—	PRS	POCI	PSS	PES	CCS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved The write value should always be 0.
20	PRSC	0	R/W (1)	Port Reset Status Change This bit indicates that port reset (bus reset) has finished. The host controller sets this bit when 10 ms port reset (bus reset) finishes. 0b: Port reset has not finished, or the PRS (Port Reset Status) bit is not changed. 1b: Port reset has finished.
19	OCIC	0	R/W (1)	Port OverCurrent Indicator Change This bit is set when the overcurrent state of the port is detected. This bit is set when the host controller changed the POCI bit value. This bit is valid only under the setting that the overcurrent state is reported for each port (OCPM bit = 1b). If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The overcurrent state has not changed. 1b: The overcurrent state has changed (POCI bit changed).
18	PSSC	0	R/W (1)	Port Suspend Status Change This bit indicates that the Resume sequence finished. This sequence includes 20 ms of the Resume signal, LS EOP, and 3 ms of resynchronization delay. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If the PRSC (Port Reset Status Change) bit is set, this bit is cleared. 0b: The Resume sequence has not finished. 1b: The Resume sequence has finished.
17	PESC	0	R/W (1)	Port Enable Status Change This bit indicates that the PES (Port Enable Status) bit was changed. If a hardware event clears the PES bit, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The PES (Port Enable Status) bit has not changed. 1b: The PES (Port Enable Status) bit has changed.

Bit	Bit Name	Initial Value	R/W	Description
16	CSC	0	R/W (1)	Connect Status Change This bit indicates that the CCS (Current Connect Status) bit was changed. If the Connect/Disconnect event occurs, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If a request (Port Reset/Port Enable/Port Suspend) is received during the Disconnect status, this bit is set for reevaluation of device connection confirmation. 0b: The CCS (Current Connect Status) bit has not changed. 1b: The CCS (Current Connect Status) bit has changed.
15 to 10	—	All 0	R	Reserved The write value should always be 0.
9	LSDA	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Low Speed Device Attached This bit indicates the speed of the device connected to the port. This bit is valid only when the CCS (Current Connect Status) bit is set. 0b: Full Speed device is connected. 1b: Low Speed device is connected. [Write] Clear Port Power This bit is used to turn off the power of the port when the port is power controlled. Writing 1b to this bit turns off the port power. Writing 0b to this bit has no effect.
8	PPS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Power Status This bit indicates the power status of the port. This bit is cleared when the overcurrent is detected. 0b: Port power is off. 1b: Port power is on. [Write] Set Port Power This bit is used to turn on the power of the port when the port is power controlled. Writing 1b to this bit turns on the port power. Writing 0b has no effect.
7 to 5	—	All 0	R	Reserved The write value should always be 0.
4	PRS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Reset Status This bit indicates the port reset (bus reset) status. When 10 ms of port reset finishes, the PRSC (Port Reset Status Change) bit is set and this bit is cleared. When the CCS bit is cleared (when no device is connected), this bit cannot be set. 0b: Not during port reset 1b: During port reset [Write] Set Port Reset This bit is used to issue a port reset (bus reset) to the downstream port. Writing 1b to this bit starts 10 ms of port reset. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to ports to which no device is connected, that port reset was performed.

Bit	Bit Name	Initial Value	R/W	Description
3	POCI	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port OverCurrent Indicator This bit indicates that the downstream port is in the overcurrent status. This bit is valid only when the setting is specified that the overcurrent status is reported for each port (OCPM bit = 1b). On the other hand, if the setting is specified that the overcurrent status is reported for all ports, this bit is always read as 0b. 0b: The port is in normal status. 1b: The port is in overcurrent status. [Write] Clear Suspend Status This bit is used to finish the Suspend status and start the Resume sequence. Writing 1b to this bit starts the Resume sequence. Writing 0b has no effect. The Resume sequence starts only when PSS (Port Suspend Status) is set.
2	PSS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Suspend Status This bit indicates that the port status is in Suspend or Resume sequence. 0b: The port is not in Suspend status. 1b: The port is in Suspend status. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is cleared under the following conditions: ● When the Resume sequence finished, and the PSSC (Port Suspend Status Change) bit is set ● When the port reset finished, and the PRSC (Port Reset Status Change) bit is set ● When the host controller is in the USB Resume state [Write] Set Port Suspend This bit is used to change the port status to Suspend. Writing 1b to this bit changes the status to Suspend. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to the ports to which no device is connected that the Suspend command was issued.
1	PES	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Enable Status This bit indicates whether the port status is Enable or Disable. If the host controller detects a bus error (for example, overcurrent state, disconnect, port power off, and babble error), it clears this bit. Then, the PESC (PortEnableStatusChange) bit is set. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is set “when port reset finished and the PRSC bit is set” or “when the port status becomes Suspend and the PRSC bit is set”. 0b: The port status is Disable. 1b: The port status is Enable. [Write] Set Port Enable This bit is used to set the PES bit. Writing 1b to this bit changes the port status to Enable. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report that the status of the ports to which no device is connected was tried to be changed to Enable.

Bit	Bit Name	Initial Value	R/W	Description
0	CCS	0	R/W	The meaning of this bit differs depending on whether the operation is read or write. [Read] Current Connect Status The current connection status of the downstream port is applied to this bit. 0b: No device is connected. 1b: A device is connected. [Write] Clear Port Enable This bit is used to clear the PES (Port Enable Status) bit. Writing 1b to this bit changes the port status to Disable. Writing 0b to this bit has no effect.

32A.2.4.2 EHCI Controller Capability Register

(1) HCIVERSION/CAPLENGTH Register

Register abbreviation: CAPL_VERSION

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Interface Version Number[15:8]								Interface Version Number[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	Capability Registers Length[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Interface Version Number	H'0110	R	This field indicates the EHCI version supported by the host controller. H'0110 is indicated because this host controller supports EHCI Rev1.1.
15 to 8	—	All 0	R	Reserved The write value should always be 0.
7 to 0	Capability Registers Length	H'20	R	This field is used as an offset that is added to the base address to find the start address of the EHCI Operational register. H'20 is indicated because the EHCI Operation register of this module starts from H'20.

(2) HCSPARAMS Register

Register abbreviation: HCSPARAMS

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Debug Port Number [3:0]				—	—	—	P_INDICATOR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	N_CC				N_PCC				Port Routing Rules	—	—	PPC	N_PORTS[3:0]			
Initial Value	0	0	0	1	0	0	0	1	1	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved The write value should always be 0.
23 to 20	Debug Port Number	All 0	R	This field indicates that the host controller port is a debug port. 0000b is indicated because this module does not have a debug port.
19 to 17	—	All 0	R	Reserved The write value should always be 0.
16	P_INDICATOR	0	R	Port Indicators This bit indicates whether the host controller supports port indicator control. 0b is indicated because this module does not support port indicator control.
15 to 12	N_CC	H'1	R	Number of Companion Controller This field indicates the number of OHCI host controllers implemented in this module. H'1 is indicated because this module has one OHCI host controller.
11 to 8	N_PCC	H'1	R	Number of Ports per Companion Controller This field indicates the number of ports supported by an OHCI host controller. The setting value of the Port_no field of the PCI Configuration EXT1 register is applied to this field.
7	Port Routing Rules	1	R	This bit indicates how individual ports are mapped to the OHCI host controller. This bit indicates 1b because, in this module, the contents of the HCSP_PORTROUTE register show the mapping method.
6, 5	—	All 0	R	Reserved The write value should always be 0.
4	PPC	1	R	Port Power Control This bit indicates how the port power of this module is controlled. 0b: The port power is always on. 1b: The PP bit of the PORTSC register controls the port power.
3 to 0	N_PORTS	H'1	R	This field indicates the number of physical downstream ports used by this module. 0001b: 1 Port

(3) HCCPARAMS Register

Register abbreviation: HCCPARAMS

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	32-Frame Periodic List Capability	Per-Port Change Event Capability	Link Power Management Capability	Hardware Prefetch Capability
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EECP								Isochronous Scheduling Threshold				—	Asynchronous Schedule Park Capability	Programmable Frame List Flag	64-bit Addressing Capability
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved The write value should always be 0.
19	32-Frame Periodic List Capability	1	R	This bit indicates 1b because this module supports 32-Frame Periodic List defined in EHCI V1.1. This means that the Frame List Size field of the USBCMD register is set to 11b, so this module supports 32-Frame Periodic List.
18	Per-Port Change Event Capability	1	R	This bit indicates 1b because this module supports event detection function for ports that are defined in EHCI V1.1. This means that this module supports event detection function for ports, and is associated with the Pre-Port Change Event Enable field of the USBCMD register, Port-1 Change Detect field of the USBSTS register, and Port-1 Change Interrupt Enable field of the USBINT register. If this bit is 0b, the corresponding fields of the above registers are treated as Reserved.
17	Link Power Management Capability	1	R	This bit indicates 1b because this module supports LPM (Link Power Management) defined in EHCI V1.1. This means that this module supports LPM L1 state, and is controlled by the Suspend using L1 bit, Suspend Status bit, and Device Address field of the PORTSC register. If this bit is 0b, the corresponding bits and field of the above PORTC register are treated as Reserved.
16	Hardware Prefetch Capability	0	R	This bit indicates 0b because this module does not support the hardware prefetch function defined in EHCI V1.1.
15 to 8	EECP	All 0	R	This field indicates the offset address of the EHCI Extend Capabilities Registers. This field indicates H'00 because this module does not use EHCI Extend Capabilities Registers.
7 to 4	Isochronous Scheduling Threshold	All 0	R	This field indicates H'0 because this module does not support caches with isochronous data structure for the entire frame.
3	—	0	R	Reserved The write value should always be 0.
2	Asynchronous Schedule Park Capability	1	R	This bit indicates whether the Park mode is supported for High Speed QH (Queue Head) in an asynchronous schedule. This bit indicates 1b because this module supports the above function.

Bit	Bit Name	Initial Value	R/W	Description
1	Programmable Frame List Flag	1	R	This bit indicates the setting for the available frame list size. This bit indicates 1b for this module. If this bit is set to 1b, bit [3:2] (Frame List Size) of the USBCMD register can be used to set the available frame list size, and the frame list size smaller than 4 Kbyte is configurable.
0	64-bit addressing Capability	0	R	This bit indicates which type of memory pointer the data structure uses (32 bit address memory pointer or 64 bit address memory pointer). This bit indicates 0b for this module because this module has the data structure that uses 32 bit address memory pointer. 64 bit address is not supported.

(4) HCSP-PORTROUTE Register

Register abbreviation: HCSP_PORTROUTE

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Companion Port Route[31:24]								Companion Port Route[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Companion Port Route[15:8]								Companion Port Route[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Companion Port Route[31:0]	All 0	R	This field indicates the ports controlled by the OHCI host controller. This field indicates 0b for this module because this module has one OHCI host controller.

32A.2.4.3 HCI Operational Register

(1) USBCMD Register

Register abbreviation: USBCMD

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	Host-Initiated Resume Duration				Interrupt Threshold Control							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Per-Port Change Events Enable	—	—	—	Asynchronous Schedule Park Mode Enable	—	Asynchronous Schedule Park Mode Count		—	Interrupt on Async Advance Doorbell	Asynchronous Schedule Enable	Periodic Schedule Enable	Frame List Size		HCRESET	RS
Initial Value	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	W	R/W

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31 to 28</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>27 to 24</td><td>Host-Initiated Resume Duration</td><td>All 0</td><td>R/W</td><td>This field indicates the minimum time of the K-state drive while the host controller resumes from the LPM (L1) state.
The value in this field is sent to the connected device that has the LPM function via the HIDR field in the bmAttributes field of the LPM token.
The encoded value of this field is defined as the name of the HIRD field in the LPM Token.
Specifically, H’0 means 50 µs, and if the value is incremented by 1, 75 µs is incremented. For example, H’1 means 125 µs, and H’4 means 1175 µs.</td></tr>
<tr><td>23 to 16</td><td>Interrupt Threshold Control</td><td>H’08</td><td>R, R/W</td><td>This field indicates the maximum rate until the host controller issues an interrupt. Note that the setting values other than the following values are not guaranteed.
<table>
<tr><th>Value</th><th>Maximum Interrupt Interval</th></tr>
<tr><td>H’00</td><td>Reserved</td></tr>
<tr><td>H’01</td><td>1 micro-frame</td></tr>
<tr><td>H’02</td><td>2 micro-frames</td></tr>
<tr><td>H’04</td><td>4 micro-frames</td></tr>
<tr><td>H’08</td><td>8 micro-frames (default, equals to 1 ms)</td></tr>
<tr><td>H’10</td><td>16 micro-frames (2 ms)</td></tr>
<tr><td>H’20</td><td>32 micro-frames (4 ms)</td></tr>
<tr><td>H’40</td><td>64 micro-frames (8 ms)</td></tr>
</table>
Do not set H’00 to this field while the Halted bit is 0b.</td></tr>
<tr><td>15</td><td>Per-Port Change Events Enable</td><td>0</td><td>R/W</td><td>This field is used to enable the event report function of the ports defined by the Port-1 Change Detect field of the USBSTS register and the Port-1 Change Detect Enable field of the USBINTR register.
0b: The event report function of the ports is disabled.
1b: The event report function of the ports is enabled.</td></tr>
<tr><td>14 to 12</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>11</td><td>Asynchronous Schedule Park Mode Enable</td><td>1</td><td>R/W</td><td>This bit enables or disables the Asynchronous Schedule Park mode.
0b: The Park mode is disabled.
1b: The Park mode is enabled.</td></tr>
</table>

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>10</td><td>—</td><td>0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>9, 8</td><td>Asynchronous Schedule Park Mode Count</td><td>11b</td><td>R/W</td><td>This field sets the number of transactions that the host controller can serially execute for one QH (Queue Head) fetch in an asynchronous schedule.
The valid value range is H’1 to H’3.
This field is valid when bit 11 (Asynchronous Schedule Park Mode Enable) is 1b.
Do not set H’0 to this field.</td></tr>
<tr><td>7</td><td>—</td><td>0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>6</td><td>Interrupt on Async Advance Doorbell</td><td>0</td><td>R/W</td><td>This bit is used as the doorbell.
In an asynchronous schedule processing, if you want an interrupt to occur before proceeding to the next QH (Queue), set this bit to 1b.
After a QH processing normally finishes, the host controller clears this bit to 0b, and sets bit 5 (Interrupt on Async Advance bit) of the USBSTS register to 1b.
If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt occurs at the next interrupt timing.
If the asynchronous schedule is disabled, do not write 1b to this bit.</td></tr>
<tr><td>5</td><td>Asynchronous Schedule Enable</td><td>0</td><td>R/W</td><td>This bit sets whether the host controller proceeds to the asynchronous list processing or skip the processing.
0b: The asynchronous list processing is skipped.
1b: Use the ASYNCLISTADDR register to proceed the asynchronous list processing.</td></tr>
<tr><td>4</td><td>Periodic Schedule Enable</td><td>0</td><td>R/W</td><td>This bit sets whether the host controller proceeds or skips the periodic list processing.
0b: The periodic list processing is skipped.
1b: Use the PERIODICLISTBASE register to proceed the periodic list processing.</td></tr>
<tr><td>3, 2</td><td>Frame List Size</td><td>0</td><td>R/W</td><td>This field specifies the frame list size.
The setting value of this field determines the size of the Frame List Current index of the FRINDEX register.<table><tr><th>Value</th><th>the number of frames in Frame List</th></tr><tr><td>00b</td><td>1024 frames (default)</td></tr><tr><td>01b</td><td>512 frames</td></tr><tr><td>10b</td><td>256 frames</td></tr><tr><td>11b</td><td>32 frames</td></tr></table></td></tr>
<tr><td>1</td><td>HCRESET</td><td>0</td><td>W</td><td>Host Controller Reset
This bit is used to initialize the EHCI circuit of the host controller.
If this bit is set to 1b, the host controller initializes the internal pipelines, counters, and state machines, and communications on the USB immediately stop. At this time, port reset is not issued to the downstream ports.
Reset by this bit has no effect on the registers other than the EHCI Operational register. The EHCI Operational register is initialized, and the port owner returns to OHCI.
Software must be reset to return the host controller to the operational state.
When this reset processing finishes, the host controller sets this bit to 0b. Writing 0b to this bit cannot stop the reset processing. If the HCHalted bit of the USBSTS register is set to 0b, do not set 1b to this bit.</td></tr>
<tr><td>0</td><td>RS</td><td>0</td><td>R/W</td><td>Run/Stop
This bit is used to run or stop the EHCI host controller.
If this bit is set to 1b, the host controller starts operation.
As long as this bit is set to 1b, the host controller continues running.
If this bit is set to 0b, the host controller finishes the currently executing transaction and some other transactions, and then is changed to the Halt status. The HCHalted bit of the USBSTS register indicates that the host controller finished the transaction processing and entered into the stop status.
If the host controller is in a status other than Halt (the HCHalted bit of the USBSTS register is 1b), do not write 1b to this bit.</td></tr>
</table>

(2) USBSTS Register

Register abbreviation: USBSTS

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Port-1 Change Detect
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Asynchronous Schedule Status	Periodic Schedule Status	Reclamation	HCHalted	—	—	—	—	—	—	Interrupt on Async Advance	Host System Error	Frame List Rollover	Port Change Detect	USBERRINT	USBINT
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)	R/W(1)

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved The write value should always be 0.
16	Port-1Change Detect	0	R/W (1)	If this bit is set to 1b, it indicates that a change in the port status was detected. This bit is used only when the Per-Port Change Events Enable bit of the USBCMD register is set to 1b.
15	Asynchronous Schedule Status	0	R	This bit indicates the current status of the asynchronous schedule. 0b: The asynchronous schedule is disabled. 1b: The asynchronous schedule is enabled. If this bit and bit 5 (Asynchronous Schedule Enable) of the USBCMD register have the same value, the asynchronous schedule is enabled (1b) or disabled (0b).
14	Periodic Schedule Status	0	R	This bit indicates the current status of the periodic schedule. 0b: The periodic schedule is disabled. 1b: The periodic schedule is enabled. If this bit and bit 4 (Periodic Schedule Enable) of the USBCMD register have the same value, the periodic schedule is enabled (1b) or disabled (0b).
13	Reclamation	0	R	This bit is used to detect an empty asynchronous schedule. If this bit is 1b, the asynchronous schedule is empty. After reset or when a QH (H = 1) is fetched, the host controller clears this bit to 0b. Also, when the host controller executes an asynchronous transaction or detects a start event, it sets this bit to 1b. If this bit is 0b and a QH (H = 1) is fetched, the host controller is entered into the Async Sched Sleeping mode.
12	HCHalted	1	R	If the Run/Stop bit of the USBCMD register is 1b, this bit indicates 0b. If the software or host controller sets the Run/Stop bit to 0b, the host controller stops operation, and sets 1b to this bit. 0b: The EHCI host controller is running. 1b: The EHCI host controller is stopped.
11 to 6	—	All 0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	Interrupt on Async Advance	0	R/W (1)	This bit indicates the Async Advance Interrupt status. After the host controller fetches the QH, it checks bit 6 (Interrupt on Async Advance Doorbell [IAAD] bit) of the USBCMD Register. If the IAAD bit is set to 1b, the host controller clears the IAAD bit after the QH processing normally finishes, and sets this bit. If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt due to this source will occur at the next interrupt timing after 1b is set to this bit. If HCD writes 1b to this bit, this bit can be cleared. Writing 0b to this bit has no effect. 0b: Async Advance Interrupt not occurred. 1b: Async Advance Interrupt status is detected.
4	Host System Error	0	R/W (1)	This bit is set to 1b if a serious error occurs in the host controller. If this error occurs, the host controller clears the Run/Stop bit of the USBCMD register to 0b so that the subsequent schedules are not executed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: No system error occurred. 1b: A system error occurred.
3	Frame List Rollover	0	R/W (1)	If a frame list rollover occurs, the host controller sets this bit to 1b. Specifically, when the Frame Index field of the FRINDEX register is returned to H’000 from the maximum value (rollover), the host controller sets this bit to 1b. The maximum value (the value at which rollover occurs) depends on the Frame List Size field of the USBCMD register. For example, if the Frame List Size is 1024 frame, rollover occurs every time FRIMDEX [13] toggles. If the Frame List Size is 512 frame, rollover occurs every time FRINDEX [12] toggles. If the Frame List Size is 256 frame, rollover occurs every time FRINDEX [11] toggles. Writing 1b to this bit can clears this bit. Writing 0b to this bit has no effect. 0b: The frame list is not returned to H’000. 1b: The frame list is returned to H’000 (rollover occurred).
2	Port Change Detect	0	R/W (1)	This bit indicates that the port status has changed. Among the ports for which the Port Owner bit of the PORTSC1 register is set to 0b, if any port satisfies one of the following conditions, the host controller sets this bit to 1b: ● Connect or Disconnect status of a device is detected, and the Connect Status Change bit of the PORTSC1 register is changed from 0 to 1. ● A change of the Enable status of the port is detected, and the Port Enable/Disable Change bit of the PORTSC1 register is changed from 0 to 1. ● The overcurrent state is detected, and the Over-current Change bit of the PORTSC1 register is changed from 0 to 1. ● J-K transition is detected on a port in the Suspend status, and the Force Port Resume bit of the PORTSC1 register is changed from 0 to 1. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect.
1	USBERRINT	0	R/W (1)	USB Error Interrupt This bit indicates that a USB transaction finished with an error. When a USB transaction finishes with an error, the host controller sets this bit to 1b. If 1b is set to the IOC bit of qTD at which an error interrupt occurred, 1b is set to both of this bit and USBINT bit. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: USB transaction is normal. 1b: USB transaction finished with an error.

Bit	Bit Name	Initial Value	R/W	Description
0	USBINT	0	R/W (1)	USB Interrupt This bit indicates that a USB transfer has finished. The host controller sets this bit to 1b if one of the following conditions is satisfied: ● A USB transfer has finished. ● A short packet is received. Even if a USB transfer finished with an error, if IOC (Interrupt On Complete) of the TD is set to 1b, this bit is set to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: A USB transfer has not finished. 1b: A USB transfer has finished.

(3) USBINTR Register

Register abbreviation: USBINTR

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Port-1 Change Event Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	Interrupt on Async Advance Enable	Host System Error Enable	Frame List Rollover Enable	Port Change Detect Enable	USB Error Interrupt Enable	USB Interrupt Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved The write value should always be 0.
16	Port-1 Change EventEnable	0	R/W	If this bit is 1b and the Port Change Detect bit of the USBSTS register is set to 1b, the host controller issues an interrupt.
15 to 6	—	All 0	R	Reserved The write value should always be 0.
5	Interrupt on Async Advance Enable	0	R/W	This bit sets whether bit 5 (Interrupt on Async Advance [IAA] bit) of the USBSTS register is enabled or disabled. If the IAA bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the IAA bit).
4	Host System Error Enable	0	R/W	This bit sets whether bit 4 (Host System Error [HSE] bit) of the USBSTS register is enabled or disabled. If the HSE bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the HSE bit).
3	Frame List Rollover Enable	0	R/W	This bit sets whether the bit 3 (Frame List Rollover [FLR] bit) of the USBSTS register is enabled or disabled. If the FLR bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the FLR bit).
2	Port Change Detect Enable	0	R/W	This bit sets whether bit 2 (Port Change Detect [PCD] bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the PCD bit).
1	USB Error Interrupt Enable	0	R/W	This bit sets whether bit 1 (USBERRINT bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBERRINT bit).

Bit	Bit Name	Initial Value	R/W	Description
0	USB Interrupt Enable	0	R/W	This bit sets whether bit 0 (IUSBINT bit) of the USBSTS register is enabled or disabled. If the USBINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBINT bit).

(4) FRINDEX Register

Register abbreviation: FRINDEX

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	Frame Index[13:8]						Frame Index[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31 to 14</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>13 to 0</td><td>Frame Index[13:0]</td><td>All 0</td><td>R/W</td><td>This field is used by the host controller to add an index to the periodic frame list. The value in this field is incremented at the end of a micro frame.
Bit [N:3] of this field is used as the Frame List Current index. This means that, before the next index arrives, the current frame list is accessed 8 times. The value for N is determined, as follows, by the setting value of bit [3:2] (Frame List Size field) of the USBCMD register.
<table>
<tr><th>Frame List Size</th><th>Number of Frames</th><th>N</th></tr>
<tr><td>00b</td><td>1024</td><td>12</td></tr>
<tr><td>01b</td><td>512</td><td>11</td></tr>
<tr><td>10b</td><td>256</td><td>10</td></tr>
<tr><td>11b</td><td>32</td><td>12</td></tr>
</table>
Access this register only when the host controller is in stop status (bit 12 [HCHalted] = 1b). The setting value of this field is applied to the SOF frame number of the SOF token.</td></tr>
</table>

(5) CTRLDSSEGMENT Register

Register abbreviation: CTRLDSSEGMENT

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CTRLDSSEGMENT[31:24]								CTRLDSSEGMENT[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CTRLDSSEGMENT[15:8]								CTRLDSSEGMENT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CTRLDSSEGMENT	All 0	R	This register is not used because this module does not support 64-bit address method. Therefore, HCD must not access this register.

(6) PERIODICLISTBASE Register

Register abbreviation: PERIODICLISTBASE

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Base Address[31:24]								Base Address[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Base Address[15:12]				—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	Base Address	All 0	R/W	This field indicates the head address of the periodic frame list on the system memory. The host controller loads the contents of this register before starting the list processing. The host controller determines the frame list to be processed by using this field and Frame Index of the FRINDEX register. Align the address of the periodic frame list by 4 Kbyte. Normal operation is not guaranteed if any of these bits are changed during operation.
11 to 0	—	All 0	R	Reserved The write value should always be 0.

(7) ASYNCLISTADDR Register

Register abbreviation: ASYNCLISTADDR

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPL[31:23]								LPL[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPL[15:8]								LPL[7:5]			—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	LPL	All 0	R/W	Link Pointer Low This field indicates the address (on the system memory) of the Asynchronous Queue Head to be processed next time. Align the address of Asynchronous Queue Head by 32 byte.
4 to 0	—	All 0	R	Reserved The write value should always be 0.

(8) CONFIGFLAG Register

Register abbreviation: CONFIGFLAG

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved The write value should always be 0.
0	CF	0	R/W	Configuration Flag This bit controls which of OHCI or EHCI is routed by the port routing control circuit by default. At the end of the host controller configuration, this bit is set to 1b. 0b: The port routing control circuit routes each port to the OHCI host controller by default. 1b: The port routing control circuit routes each port to the EHCI host controller by default.

(9) PORTSC1Register

Register abbreviation: PORTSC1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Device Address							Suspend Status[1]	Suspend Status[0]	WKOC_E	WKDSCNNT_E	WKCNNT_E	Port Test Control			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	Port Owner	PP	Line Status		Suspend using L1	Port Reset	Suspend	Force Port Resume	Over current Active Change	Over-current Active	Port Enabled /Disabled Change	Port Enabled /Disabled	Connect Status Change	Current Connect Status
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W(1)	R	R/W(1)	R/W	R/W(1)	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	Device Address	All 0	R/W	This field indicates the USB device address (7 bits) of the device connected to the down port. This address is used when the LPM Token is sent. If the value of this filed is H’00, it means that no device that needs to use this field has been connected.
24, 23	Suspend Status	All 0	R	This field indicates the response from the connected device to the LPM Token (L1 transition request). 00b: The device succeeded transition to the L1 state (ACK received from the device). 01b: The device has not changed to the L1 state (NYET received from the device). 10b: The device does not support the L1 state transition (STALL received from the device). 11b: Other response (for example, Timeout error) Change this field only when the Suspend bit is 0b.
22	WKOC_E	0	R/W	Wake on Overcurrent Enable By writing 1b to this bit, the overcurrent state can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
21	WKDSCNNT_E	0	R/W	Wake on Disconnect Enable By writing 1b to this bit, device disconnection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
20	WKCNNT_E	0	R/W	Wake on Connect Enable By writing 1b to this bit, device connection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>19 to 16</td><td>Port Test Control</td><td>All 0</td><td>R/W</td><td>This field is controlled by the test mode.
If the value of this field is other than 0000b, it indicates that this module is running in the test mode.
<table>
<tr><th>Value</th><th>Test mode</th></tr>
<tr><td>0000b</td><td>Test mode not enabled</td></tr>
<tr><td>0001b</td><td>Test J_STATE</td></tr>
<tr><td>0010b</td><td>Test K_STATE</td></tr>
<tr><td>0011b</td><td>Test SE0_NAK</td></tr>
<tr><td>0100b</td><td>Test Packet</td></tr>
<tr><td>0101b</td><td>Test FORCE_ENABLE</td></tr>
<tr><td>Other</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>15, 14</td><td>—</td><td>All 0</td><td>R</td><td>Reserved
The write value should always be 0.</td></tr>
<tr><td>13</td><td>Port Owner</td><td>1</td><td>R/W</td><td>This bit indicates which of OHCI or EHCI has the port ownership.
0b: EHCI has the port ownership.
1b: OHCI has the port ownership.
When bit 0 (Configure Flag bit) of the CONFIGFLA register is changed from 0b to 1b, this bit becomes 0b. If bit 0 (Configure Flag bit) of the CONFIGFLA register is 0b, this bit becomes 1b.
If the connected device is not a High Speed device, this bit is set to 1b to transfer the port ownership to OHCI.</td></tr>
<tr><td>12</td><td>PP</td><td>0</td><td>R/W</td><td>Port Power
This bit controls power supply to the port.
If this bit is 0b, power is not supplied to the port. Therefore, the port does not function, and does not recognize connection and disconnection.
If overcurrent is detected while this bit is set to 1b, the host controller clears this bit to 0b, and the power supplied to the port is stopped.
<i>Note</i>: As described later, if the PPC bit is 0b, this bit is fixed to 1b, so the power supplied to the port is not stopped.
The function of this bit differs depending on the value of bit 4 (PPC [Port Power Control] bit) of the HCSPARAMS register.
<table>
<tr><th>PPC</th><th>PP</th><th colspan="3">Operation</th></tr>
<tr><td>0b</td><td>1b</td><td>R</td><td colspan="2">This bit is fixed to 1b, and the power is always supplied to the port.</td></tr>
<tr><td rowspan="3">1b</td><td rowspan="3">0b/1b</td><td rowspan="3">R/W</td><td colspan="2">Whether or not power is supplied to the port depends on the setting of this bit.</td></tr>
<tr><td>0b</td><td>Power is not supplied to the port.</td></tr>
<tr><td>1b</td><td>Power is supplied to the port.</td></tr>
</table></td></tr>
<tr><td>11, 10</td><td>Line Status</td><td>All 0</td><td>R</td><td>This field indicates the logical level of D+/D- lines of the current USB bus. (bit 11: DP / bit 10: DM)
This field is used to detect an LS device before starting a sequence for port reset and port enable.
Therefore, this bit is valid only when bit 3 (Port Enable/Disable bit) is 0b and bit 0 (Current Connect Status bit) is 1b.
<table>
<tr><th>bit 11</th><th>bit 10</th><th>USB bus status</th><th>Description</th></tr>
<tr><td rowspan="2">0b</td><td rowspan="2">0b</td><td rowspan="2">SE0</td><td>The device is not an LS device.</td></tr>
<tr><td>EHCI port reset is executing.</td></tr>
<tr><td rowspan="2">1b</td><td rowspan="2">0b</td><td rowspan="2">J-state</td><td>The device is not an LS device.</td></tr>
<tr><td>EHCI port reset is executing.</td></tr>
<tr><td rowspan="2">0b</td><td rowspan="2">1b</td><td rowspan="2">K-state</td><td>An LS device was connected.</td></tr>
<tr><td>The port ownership is transferred from EHCI to OHCI.</td></tr>
<tr><td rowspan="2">1b</td><td rowspan="2">1b</td><td rowspan="2">Undefined</td><td>The device is not an LS device.</td></tr>
<tr><td>EHCI port reset is executing.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>9</td><td>Suspend using L1</td><td>0</td><td>R/W</td><td>Suspend using L1 (LPM) control bit.
If 1b is written to the Suspend bit (bit 7) while this bit is 1b, the state of this module is changed to the LPM state.
Writing to this bit is possible only when the Suspend bit (bit 7) is 0b.
0b: Suspend using L2
1b: Suspend using L1 (LPM)
If this bit is 1b and the Device Address field is other than H’0000, when the Suspend bit is set to 1b, the host controller generates an LPM Token to change the state to the L1 state.
If this bit is 0b, the host controller operates as L2 Suspend.</td></tr>
<tr><td>8</td><td>Port Reset</td><td>0</td><td>R/W</td><td>This bit indicates the reset status of the port.
0b: The port is not being reset.
1b: The port is being reset.
If 1b is written to this bit while this bit is 0b, the bus reset sequence defined in USB 2.0 standard starts. To finish the bus reset sequence, 0b must be written to this bit. Note that this bit must remain 1b long time enough to guarantee that the bus reset sequence defined in USB 2.0 standard will be complete.
If bit 12 (HCHalted) of the USBSTS register is 1b, do not set this bit to 1b.
If any of the PP (Port Power) bit, Port Owner bit, and Current Connect Status bit is in the following status, this bit becomes 0b.
Note: Even if 1b is written to this bit, the bus reset sequence does not start.
● PP (Port Power) bit = 0b
● Port Owner bit = 1b
● Current Connect Status bit = 0b</td></tr>
<tr><td>7</td><td>Suspend</td><td>0</td><td>R/W</td><td>This bit indicates the Suspend control and status of the port.
This bit and bit 2 (Port Enabled/Disabled bit) indicate the following port status.
<table>
<tr><th>Port Enabled</th><th>Suspend</th><th>Port Status</th></tr>
<tr><td>0b</td><td>—</td><td>Disable</td></tr>
<tr><td>1b</td><td>0b</td><td>Enable</td></tr>
<tr><td></td><td>1b</td><td>Suspend</td></tr>
</table>
To change the port state to L1 or L2 Suspend, set this bit to 1b.
Whether the host controller supports L1 Suspend state or L2 Suspend state depends on the value in the Suspend Using L1 bit.
In the Suspend state, data transfer to the downstream port is blocked by this port (except for port reset). If this bit is set to 1b during data transfer, blocking of transfer data does not occur until the current data transfer finishes.
Writing 0b to this bit has no effect.
This bit can be set to 1b only when all the following conditions are satisfied: “PP (Port Power) bit = 1b”, “Port Owner bit = 0b”, and “Current Connect Status bit = 1b”. If any of the following conditions is satisfied, the host controller clears this bit:
● “Resume is complete” is detected.
● The PR bit is set to 1b when PR (Port Reset) bit is 0b.
● The port owner is 1b (OHCI).
● The PP (Port Power) bit is set to 0b.
● The Port Enabled/Disabled bit is set to 0b.</td></tr>
</table>

Bit	Bit Name	Initial Value	R/W	Description
6	Force Port Resume	0	R/W	This bit indicates that the Resume state of the port is detected. 0b: Resume signal is not detected or output. 1b: Resume (K-state) is detected or output. When the port is in Suspend state, if the host controller detects transition of the state from J to K (if RemoteWakeup is detected from the connected device), it sets this bit to 1b. The host controller also sets the Port Change Detect bit or Port-1 Changes Detect bit of the USBSTS register to 1b. If this bit is set to 1b, the host controller does not set 1b to the Port Change Detect bit and Port-1 Changes Detect bit of the USBSTS register. While this bit is 1b, the Resume signal (FS K) is driven onto the USB bus. For L2 transition, this bit must be cleared to 0b after an appropriate time has passed. By writing 0b to this bit while this bit is 1b, the port status is recovered to be the HS Idle status. This bit remains 1b until the port is recovered. The host controller must finish transition to the HS Idle state within 2msec since this bit is cleared to 0b. On the other hand, for L1 transition, the host controller sends a Resume signal at necessary timing, and this bit is cleared to 0b at Resume recovery. Note that the software sets the length of the Resume signal driven by the host controller, by using the Host-Initiated Resume Duration field of the USBCMD register. If the Port Power (PP) bit is 0b, this bit becomes 0b.
5	Over-current Change	0	R/W (1)	This bit indicates that bit 4 (Over-current Active bit) has changed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: Over-current Active bit has not changed. 1b: Over-current Active bit has changed.
4	Over-current Active	0	R	This bit indicates the overcurrent status of the port. If the host controller detects overcurrent, it disables the port, and set this bit to 1b. After the overcurrent state is released, the host controller automatically clears this bit from 1b to 0b. 0b: The port is not in overcurrent state. 1b: The port is in overcurrent state.
3	Port Enable/ Disable Change	0	R/W (1)	This bit indicates that the host controller detected frame babble. If the Host Controller detects frame babble, it disables the port and sets this bit to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: Frame babble has not occurred. 1b: Frame babble is detected.
2	Port Enabled/ Disabled	0	R/W	This bit indicates the Enable/Disable status of the port. The host controller resets the port, and enables the port if the connected device is recognized as an HS device, and sets this bit to 1b. The software cannot set this bit to 1b. If the host controller detects disconnection of a device or other errors, it disables the port, and clears this bit to 0b. The port also becomes disabled when 0b is written to this bit. If the port is disabled, data transfer to the downstream port is blocked except for port reset. If Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the port becomes enabled, and this bit is set to 1b. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The port is disabled. 1b: The port is enabled.
1	Connect Status Change	0	R/W (1)	This bit indicates that bit 0 (Current Connect Status bit) has changed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The Current Connect Status bit has no change. 1b: The Current Connect Status bit has changed.

Bit	Bit Name	Initial Value	R/W	Description
0	Current Connect Status	0	R	This bit indicates the connection status of the port. If the host controller detects connection of a device, it sets this bit to 1b. Also, if Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the Host Controller sets this bit to 1b even if no device is connected. On the other hand, if the host controller detects disconnection of a device, it sets this bit to 0b. If the Port Power (PP) bit is 0b, or the Port Owner (PO) bit is 0b, this bit becomes 0b. 0b: No device is connected to the port. 1b: A device is connected to the port.

32A.2.4.4 AHB Bridge Register

(1) INT_ENABLE Register

Abbreviated name of register: INT_ENABLE

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	WAKEON_INTEN	UCOM_INTEN	USBH_INTBEN	USBH_INTAEN	AHB_INTEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved The write value should always be 0.
4	WAKEON_INTEN	0	R/W	This bit enables or disables bit 4 (WAKEON_INT) of the INT_STATUS register. 0b: disable 1b: enable
3	UCOM_INTEN	0	R/W	This bit enables or disables bit 3 (UCOM_INT) of the INT_STATUS register. 0b: disable 1b: enable
2	USBH_INTBEN	0	R/W	This bit enables or disables bit 2 (USBH_INTB) of the INT_STATUS register. 0b: disable 1b: enable
1	USBH_INTAEN	0	R/W	This bit enables or disables bit 1 (USBH_INTA) of the INT_STATUS register. 0b: disable 1b: enable
0	AHB_INTEN	0	R/W	This bit enables or disables bit 0 (AHB_INT) of the INT_STATUS register. 0b: disable 1b: enable

(2) INT_STATUS Register

Abbreviated name of register: INT_STATUS

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	WAKEON_INT	UCOM_INT	USBH_INTB	USBH_INTA	AHB_INT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R	R	R	R/W(1)

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved The write value should always be 0.
4	WAKEON_INT	0	R/W (1)	This bit indicates the state of WAKEON interrupt from the HOST module. Writing 1b to this bit can clear this bit. 0b: No WAKEON interrupt 1b: WAKEON interrupt
3	UCOM_INT	0	R	This bit indicates the state of interrupt from the UCOM register. To clear the interrupt, use the UCOM2 Register. 0b: No UCOM register interrupt 1b: UCOM register interrupt
2	USBH_INTB	0	R	This bit indicates the state of EHCI interrupt. To clear the interrupt, use the USBSTS Register (of the EHCI Operational Register). 0b: No INTB interrupt 1b: INTB interrupt
1	USBH_INTA	0	R	This bit indicates the state of OHCI interrupt. To clear the interrupt, use the HcInterruptStatus Register (of the OHCI Operational Register). 0b: No INTA interrupt 1b: INTA interrupt
0	AHB_INT	0	R/W (1)	This bit indicates that a BUS Master error occurred. Writing 1b to this bit can clear this bit. 0b: No bus error occurred. 1b: A bus error occurred.

(3) AHB_BUS_CTR Register

Abbreviated name of register: AHB_BUS_CTR

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT_TYPE				—	—	—	PROT_MODE	—	—	ALIGN_ADDRESS		—	—	MAX_BURST_LEN	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved The write value should always be 0.
15 to 12	PROT_TYPE	All 0	R/W	This field sets MHPROT [3:0] used when the BUS Master interface issues a transfer request. bit 15 0b: Cache disabled. 1b: Cache enabled. bit 14 0b: Buffer disabled. 1b: Buffer enabled. bit 13 0b: User access 1b: Privileged access bit 12 0b: Operation code 1b: Data
11 to 9	—	All 0	R	Reserved The write value should always be 0.
8	PROT_MODE	0	R/W	This bit selects the mode of MHPROT [3:0] used when the Master interface issues a transfer request. 0b: The value of PROT_TYPE is output as MHPROT [3:0]. 1b: When a DMA transfer is performed, MHPROT [3:0] is set to 0000b if the final burst is performed, or MHPROT [3:0] is set to the PROT_TYPE value if another burst transfer is performed.
7, 6	—	All 0	R	Reserved The write value should always be 0.
5, 4	ALIGN_ADDRESS	All 0	R/W	This field sets the address boundary used when the BUS Master interface issues a burst transfer. 00b: A burst transfer is issued so that not to exceed 1-Kbyte boundary. 01b: A burst transfer is issued so that not to exceed 64-byte boundary. 10b: A burst transfer is issued so that not to exceed 32-byte boundary. (The maximum burst length is INCR8. This is because, if INCR16 is used, 32-byte boundary is exceeded.) 11b: A burst transfer is issued so that not to exceed16 byte boundary. (The maximum burst length is INCR4. This is because, if the length is at least INCR8, 16-byte boundary is exceeded.)
3,2	—	All 0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	MAX_BURST _LEN	All 0	R/W	This field selects the maximum burst length used when the BUS Master interface issues a transfer request. 00b: INCR16 01b: INCR8 10b: INCR4 11b: SINGLE

(4) USBCTR Register

Abbreviated name of register: USBCTR

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DIRPD	PLL_RST	USBH_RST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved The write value should always be 0.
2	DIRPD	0	R/W	USBPHY Standby Mode Control 0b: USBPHY normal operating mode 1b: USBPHY standby mode *Note*: When the USB module is not in use, setting this bit to 1 reduces power consumption by the USBPHY module. Only set it to 1 when the USB module is not in use. In transitions from USBPHY standby mode to USBPHY normal operating mode, assert the reset signal for the USBPHY module for at least 1 μs before the transition.
1	PLL_RST	1	R/W	This bit controls resetting of the USBPHY module. 0b: The USBPHY reset is released. 1b: The USBPHY module is reset.
0	USBH_RST	0	W	Software reset to this module. Setting this bit to 1b resets this module entirely. This bit is always read as 0b. *Note*: Set this bit only when the BUS Master interface of this module is not running. Access to this module becomes valid 10 CLK (internal bus clock [P1ϕ]) after this bit is written. 0b: Nothing occurs. 1b: Reset is issued to this module.

(5) Register Enable/Clock Gating Control Register

Abbreviated name of register: REGEN_CG_CTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NONUSE_CLK_MSK	—	HOST_CLK_MSK	PERI_CLK_MSK	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	NONUSE_CLK_MSK	0	R/W	This bit is used to mask the clock for the unused host or peripheral controller, depending on the setting value of the OTG_PERI bit of the COMMCTRL register. 0b: Do not mask the clock. 1b: Mask the clock. For details, see **Section 32A.3.1.2, Specifications of NONUSE_CLK_MSK operation**.
30	—	0	R	Reserved The write value should always be 0.
29	HOST_CLK_MSK	0	R/W	This bit is used to forcibly mask clock supply to the host controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see **Section 32A.3.1.3, Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations**.
28	PERI_CLK_MSK	0	R/W	This bit is used to forcibly mask the clock supply to the peripheral controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see **Section 32A.3.1.3, Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations**.
27 to 24	—	All 0	R	Reserved The write value should always be 0.
23 to 0	—	All 0	R	Reserved The write value should always be 0.

(6) Suspend Control Register

Abbreviated name of register: SPD_CTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SUSPENDM_ENABLE	SLEEPM_ENABLE	—	—	—	—	—	—	WKCNNT_ENABLE	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GLOBAL_SUSPENDM_P1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	SUSPENDM_ENABLE	0	R/W	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the suspend status (the state in which PHY built-in PLL is stopped) when this module is changed to the USB Suspend mode. If this bit is set to 1b, the Suspend related bits of the OHCI/EHCI Operational registers below are set. If this module is changed to the USB Suspend mode, USBPHY is placed into the suspend status. [This function is valid for the following OHCI/EHCI Operational registers] EHCI: bit [7] (Suspend bit) of the PORTSC1 register OHCI: bit [2] (PSS bit) of the HcRhPortStatus register OHCI: bit [7:6] (HCFS field) of the HcControl register
30	SLEEPM_ENABLE	0	R/W	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the sleep status when the LPM function is used to change to the L1 Suspend mode. (In the sleep status, the PHY built-in PLL is running, but the 60MHz clock from USBPHY is gated.) This bit is valid when "Suspend using L1" of the POTSC register is 1b. If this bit is set to 1b, an L1 transition request is issued to the device. If the request is accepted, USBPHY is placed into the sleep status after the following EHCI register bit is set to 1b. [This function is valid for the following OHCI/EHCI Operational register] EHCI: bit [7] (Suspend bit) of the PORTSC1 register
29 to 24	—	All 0	R	Reserved The write value should always be 0.
23	WKCNNT_ENABLE	0	R/W	The setting of this bit is valid only in the Host mode (the OTG_PERI bit of the COMMCTRL register is 0b). If this bit is set to 1b, when a device disconnect occurs while USBPHY is in the suspend or sleep status, the suspend or sleep status is released. This bit is valid only when the SUSPENDM_ENABLE bit (bit 31) or SLEEP_ENABLE bit (bit 30) is 1b. *Note*: To set 1b to the SUSPENDM_ENABLE bit (bit 31) or SLEEPM_ENABLE bit (bit 30), as the general rule, set this bit to 1b. If, in the above case, this bit is not set to 1b, the suspend or sleep status cannot be released even if a device disconnect occurs.
22 to 1	—	All 0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	GLOBAL_SUSPENDM_P1	0	R/W	The setting of this bit is valid regardless of the value of the OTG_PERI bit of the COMMCTRL register. This bit is used to forcibly place USBPHY into the suspend status (in which the PHY built-in PLL is stopped). If this bit is set to 1b, USBPHY is placed into the suspend status, regardless of the operating status and port status of the host controller. *Note*: – Do not set this bit to 1b during data transfer. We recommend that you set this bit to 1b after “stopping the EHCI/OHCI list processing” and “placing the port in the disabled status”. – If the SUSPENDM_ENABLE bit (bit 31) is 1b, do not set this bit to 1b.

(7) Suspend/Resume Timer Setting Register

Abbreviated name of register: SPD_RSM_TIMSET

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIMER_CONNECT[15:8]								TIMER_CONNECT[7:0]							
Initial Value	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIMER_RESUME[15:8]								TIMER_RESUME[7:0]							
Initial Value	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TIMER_CONNECT[15:0]	H’01F4	R/W	This field indicates the timer value used by USBPHY to detect Device Connect/Disconnect of the device in the suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether Connect/Disconnect occurs is judged by the internal bus clock (P1ϕ). According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes at least 2.5 μs. 1 bit = 1 cycle (μs) (Setting guideline) For 100 MHz: At least H’FA
15 to 0	TIMER_RESUME[15:0]	H’03E8	R/W	This field indicates the timer value used by USBPHY to detect RemoteWakeup signal from the device in suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether the RemoteWakeup signal or not is judged by the internal bus clock (P1ϕ). According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes at least 5 μs. 1 bit = 1 cycle (μs) (Setting guideline) For 100 MHz: At least H’1F4

(8) Overcurrent Detection/Sleep Timer Setting Register

Abbreviated name of register: OC_SLP_TIMSET

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	TIMER_SLEEP[8:4]					TIMER_SLEEP[3:0]				TIMER_OC[19:16]			
Initial Value	0	0	0	0	1	1	0	0	1	0	0	0	0	0	1	1
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIMER_OC[15:8]								TIMER_OC[7:0]							
Initial Value	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved The write value should always be 0.
28 to 20	TIMER_SLEEP[8:0]	H'0C8	R/W	This field indicates the timer value used by USBPHY when the SLEEPM_ENABLE bit of the SPD_CTRL register is set to 1b. This timer value is used to measure the time for detecting RemoteWakeup reception and Resume-K drive time during sleep status (in which the PHY built-in PLL is running, but 60MHz clock from the USBPHY is gated). When USBPHY is in sleep status, whether the RemoteWakeup signal or not is judged by the internal bus clock (P1ϕ). According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes 1 µs. 1 bit = 1 cycle (µs) (Setting guideline) For 100 MHz: H'064
19 to 0	TIMER_OC [19:0]	H'30D40	R/W	This field indicates the timer value used for overcurrent detection. If the overcurrent input (OVRCUR) set in this field is continuously asserted (0b) for the duration set in this register, this module determines that overcurrent occurred. According to the internal bus clock (P1ϕ) frequency, specify the setting so that this timer value becomes at least 1 ms. 1 bit is 1 cycle (µs). (Setting guideline) For 100 MHz and 1 ms: At least H'1_86A0

(9) SBRN_FLADJ_PORTWAKECAP Register

Abbreviated name of register: SBRN_FLADJ_PW

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PORTWAKECAP[15:8]								PORTWAKECAP[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLADJ[7:0]								SBRN[7:0]							
Initial Value	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PORTWAKE CAP[15:0]	H'0003	R/W	This field is used to mask the ports (of the connected device) that are used for Wakeup event. Operation on this field has no effect on the operation of the HOST module.
15 to 8	FLADJ[7:0]	H'20	R, R/W	This field adjusts the length of one micro frame by 16HS bit time unit. The initial value indicates H'20 (60000d HS bit time).
7 to 0	SBRN[7:0]	H'20	R	This field indicates the Serial Bus Release Number. The fixed value "H'20" is indicated.

(10) PORT_LPM_CTR1 Register

Abbreviated name of register: PORT_LPM_CTRL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	NYET_RETRY_CNT_P1[3:0]				REMOTEWAKE_EN_P1	SLEEP_INT_EN_P1	RETRY_ENABLE_NYET_P1	HIRD_SEL_P1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The write value should always be 0.
7 to 4	NYET_RETRY_CNT_P1[3:0]	All 0	R/W	This field sets the number of retries that are allowed when the response from the device in the LPM transaction was NYET. The setting value of this bit is valid if the RETRY_ENABLE_NYET_P1 bit (bit 1) is 1b. 0000b: No retry. 0001b to 1111b: Retries the set number of times. (MAX: 15 retries)
3	REMOTEWAKE_EN_P1	0	R/W	This bit is used to indicate the value of the RemoteWakeup bit of the LPM Token. 0b: RemoteWakeup is supported. 1b: RemoteWakeup is not supported.
2	SLEEP_INT_EN_P1	0	R/W	This bit is used to enable an interrupt to occur when a status other than ACK is received in the LPM transaction. The Per-Port Change interrupt can occur. 0b: No interrupt occurs. 1b: An interrupt occurs.
1	RETRY_ENABLE_NYET_P1	0	R/W	This bit is used to set the behavior of the host controller when a response from the device is NYET in an LPM transaction. 0b: No retry. 1b: Retries are made.

Bit	Bit Name	Initial Value	R/W	Description
0	HIRD_SEL_P1	0	R/W	This bit sets the time for K drive for when recovered from the Sleep state. Based on the setting values of this bit and EHCI USBCMD Register bit [27:24], the K drive time is determined as follows.

USBCMD Register	HIRD_SEL_P1 (Setting value of this bit)	
bit [27:24]	0b	1
0000b	75 µs	50 µs
0001b	100 µs	125 µs
0010b	150 µs	200 µs
0011b	250 µs	275 µs
0100b	350 µs	350 µs
0101b	450 µs	425 µs
0110b	950 µs	500 µs
0111b	1950 µs	575 µs
1000b	2950 µs	650 µs
1001b	3950 µs	725 µs
1010b	4950 µs	800 µs
1011b	5950 µs	875 µs
1100b	6950 µs	950 µs
1101b	7950 µs	1025 µs
1110b	8950 µs	1100 µs
1111b	9950 µs	1175 µs

32A.2.4.5 UCOM Register

(1) Common Control Register

Abbreviated name of register: COMMCTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTG_PERI	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	OTG_PERI	1	R/W	This bit specifies whether this module is set to the host mode or function mode. 0b: Host mode 1b: Function mode (Only CH0 can be set)
30 to 0	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.

(2) OTG-BC Interrupt Status Register

Abbreviated name of register: OBINTSTA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DPMONCHG_STA	DMMONCHG_STA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W(1)	R/W(1)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CHGDETCHG1_STA	—	PDDETCHG1_STA	—	—	OCINT_STA	IDCHG_STA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R/W(1)	R	R/W(1)	R	R	R/W(1)	R/W(1)

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17	DPMONCHG_STA	0	R/W (1)	This bit is set if the DPMON bit of the LINECTRL1 register has changed. 0b: The DPMON bit of the LINECTRL1 register has not changed. 1b: The DPMON bit of the LINECTRL1 register has changed.
16	DMMONCHG_STA	0	R/W (1)	This bit is set if the DMMON bit of the LINECTRL1 register has changed. 0b: The DMMON bit of the LINECTRL1 register has not changed. 1b: The DMMON bit of the LINECTRL1 register has changed.
15 to 7	—	All 0	R	Reserved The write value should always be 0.
6	CHGDETCHG1_STA	0	R/W (1)	This bit is set if the CHGDETSTS bit of the BCCTRL1 register has changed from 0b to 1b. 0b: The CHGDETSTS bit of the BCCTRL1 register has not changed from 0b to 1b 1b: The CHGDETSTS bit of the BCCTRL1 register has changed from 0b to 1b.
5	—	0	R	Reserved The write value should always be 0.
4	PDDETCHG1_STA	0	R/W (1)	This bit is set if the PDDETSTS bit of the BCCTRL1 register has changed from 0b to 1b. 0b: The PDDETSTS bit of the BCCTRL1 register has not changed from 0b to 1b. 1b: The PDDETSTS bit of the BCCTRL1 register has changed from 0b to 1b.
3	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	OCINT_STA	0	R/W (1)	This bit is set if the OVRCUR pin is asserted. 0b: The OVRCUR pin is not asserted (and remains 1b). 1b: The OVRCUR pin is asserted (and became 0b).
0	IDCHG_STA	1	R/W (1)	This bit is set if the input value from the OTG_ID pin has changed. 0b: There is no change in the OTG_ID pin. 1b: There is a change in the OTG_ID pin. *Note*: The initial value is 1. Before using this bit, clear the status.

(3) OTG-BC Interrupt Enable Register

Abbreviated name of register: OBINTEN

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DPMONCHG_EN	DMMONCHG_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CHGDETCHG1_EN	—	PDDETCHG1_EN	—	—	OCINT_EN	IDCHG_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved The write value should always be 0.
17	DPMONCHG_EN	0	R/W	DPMONCHG_STA bit interrupt enable 0b: Interrupt via the DPMONCHG_STA bit is disabled. 1b: Interrupt via the DPMONCHG_STA bit is enabled.
16	DMMONCHG_EN	0	R/W	DMMONCHG_STA bit interrupt enable 0b: Interrupt via the DMMONCHG_STA bit is disabled. 1b: Interrupt via the DMMONCHG_STA bit is enabled.
15 to 7	—	All 0	R	Reserved The write value should always be 0.
6	CHGDETCHG1_EN	0	R/W	CHGDETCHG1_STA bit interrupt enable 0b: Interrupt via the CHGDETCHG1_STA bit is disabled. 1b: Interrupt via the CHGDETCHG1_STA bit is enabled.
5	—	0	R	Reserved The write value should always be 0.
4	PDDETCHG1_EN	0	R/W	PDDETCHG1_STA bit interrupt enable 0b: Interrupt via the PDDETCHG1_STA bit is disabled. 1b: Interrupt via the PDDETCHG1_STA bit is enabled.
3	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	OCINT_EN	0	R/W	OCINT_STA bit interrupt enable 0b: Interrupt via the OCINT_STA bit is disabled. 1b: Interrupt via the OCINT_STA bit is enabled.
0	IDCHG_EN	0	R/W	IDCHG_STA bit interrupt enable 0b: Interrupt via the IDCHG_STA bit is disabled. 1b: Interrupt via the IDCHG_STA bit is enabled.

(4) VBUS Control Register

Abbreviated name of register: VBCTRL

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	VGPUO	—	—	—	VBOUT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved The write value should always be 0.
29	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
28 to 22	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
21	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
20 to 17	—	All 0	R	Reserved The write value should always be 0.
16	—	1	R	Reserved The write value should always be 1.
15 to 5	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
4	VGPUO	0	R/W	The level corresponding to the inverse of this bit (in terms of positive logic) is output from the OTG_EXICEN pin. This bit is used, for example, for control of the external power IC.
3 to 1	—	All 0	R	Reserved The write value should always be 0.
0	VBOUT	0	R/W	This bit is one of the VBUSEN control bits. This bit is used to assert VBUS by controlling the external power IC. 0b: VBUS output disable 1b: VBUS output enable If overcurrent occurs, this bit is automatically cleared to 0.

(5) Line Control Port 1 Register

Abbreviated name of register: LINECTRL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	DPRPU_EN	DP_RPU	DPRPD_EN	DP_RPD	DMRPD_EN	DM_RPD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	DPMON	DMMON	—	IDMON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved
25, 24	—	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23, 22	—	All 0	R	Reserved
21	DPRPU_EN	0	R/W	This bit enables the DP_RPU (bit 20) setting for pullup control of D+ in port 1. 0: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is disabled. 1: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is enabled.
20	DP_RPU	0	R/W	This bit controls pullup of D+ in port 1. This control is enabled only when DPRPU_EN (bit 21) = 1. Note that if this bit is cleared to 0 when DPRPU_EN (bit 21) = 1, D+ is in the floating state while no device drives the USB bus. Note also that setting DPRPD_EN (bit 19) and DP_RPD (bit 18) to enabled is prohibited while this bit is set to enabled - that is, enabling both pullup and pulldown of D+ together is prohibited. 0: Pullup of D+ in port 1 is disabled. 1: Pullup of D+ in port 1 is enabled.
19	DPRPD_EN	0	R/W	This bit enables DP_RPD (bit 18) to control USB bus (DP) 15 kΩ Pulldown resister. 0b: Control of DP-side 15 kΩ Pulldown resister by DP_RPD (bit 18) is disabled. 1b: Control of DP-side 15 kΩ Pulldown resister by DP_RPD (bit 18) is enabled.
18	DP_RPD	0	R/W	This bit controls USB bus (DP) 15 kΩ Pulldown resister when DPRPD_EN (bit 19) = 1b. 0b: DP-side 15 kΩ Pulldown resister is OFF. 1b: DP-side 15 kΩ Pulldown resister is ON.
17	DMRPD_EN	0	R/W	This bit enables DM_RPD (bit 16) to control USB bus (DM) 15 kΩ Pulldown resister. 0b: Control of DM-side 15 kΩ Pulldown resister by DM_RPD (bit 16) is disabled. 1b: Control of DM-side 15 kΩ Pulldown resister by DM_RPD (bit 16) is enabled.
16	DM_RPD	0	R/W	This bit controls USB bus (DM) 15 kΩ Pulldown resister when DMRPD_EN (bit 17) = 1b. 0b: DM-side 15 kΩ Pulldown resister is OFF. 1b: DM-side 15 kΩ Pulldown register is ON.
15 to 4	—	All 0	R	Reserved The write value should always be 0.
3	DPMON	0	R	This bit indicates the value of USB bus DP.
2	DMMON	0	R	This bit indicates the value of USB bus DM.
1	—	0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	IDMON	0	R	This bit indicates the value of the OTG_ID pin.

(6) BC Control Port 1 Register

Abbreviated name of register: BCCTRL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PDDETSTS	CHGDETSTS	—	—	DCPMODE	VDMSRCE	IDPSINKE	VDPSRCE	IDMSINKE	IDPSRCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	H’00_C000	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
9	PDDETSTS	0	R	This bit indicates the USBPHY Portable Device Detect signal state.
8	CHGDETSTS	0	R	This bit indicates the USBPHY Charging Downstream Port Detect signal state.
7, 6	—	All 0	R	Reserved The write value should always be 0.
5	DCPMODE	0	R/W	If USBPHY is used as DCP (Dedicated Charging Port), this bit is set to 1b.
4	VDMSRCE	0	R/W	This bit controls the USBPHY built-in VDM_SRC circuit. If this bit is set to 1b, VDM_SRC goes ON, and the DM pin is driven.
3	IDPSINKE	0	R/W	This bit controls the USBPHY built-in Portable Device Detect circuit. If this bit is set to 1b, Portable Device detection is enabled.
2	VDPSRCE	0	R/W	This bit controls the USBPHY built-in VDP_SRC circuit. If this bit is set to 1b, VDP_SRC goes ON, and the DP pin is driven.
1	IDMSINKE	0	R/W	This bit controls the USBPHY built-in Charging Downstream Port Detect circuit. If this bit is set to 1b, Charging Downstream Port detection is enabled.
0	IDPSRCE	0	R/W	This bit controls the USBPHY built-in IDP_SRC circuit. If this bit is set to 1b, IDP_SRC goes ON, and the DP pin is driven.

32A.3 Clock Signals

32A.3.1 Clock Gating Specifications

32A.3.1.1 Overview of clock gating

Because this module has a feature of switching the host controller and peripheral controller, clock supply to the unused controller might not be necessary.

Therefore, three clock gating control bits are allocated in the Register Enable/Clock Gating Control Register for the purpose of reducing power consumption by implementing clock gating for the “circuits that do not require clock supply temporarily.”

The clock gating control bits can be controlled to gate the clock that is supplied to the host controller or the peripheral controller.

[Target Register]

Register Enable/Clock Gating Control Register (offset:H’304)

[Functional specification]

bit	Symbol	Functional specification
31	NONUSE_CLK_MSK	Gating clocks for unused host controller or peripheral controller.
29	HOST_CLK_MSK	Gating clocks for host controller.
28	PERI_CLK_MSK	Gating clocks for peripheral controller.

32A.3.1.2 Specifications of NONUSE_CLK_MSK operation

This function automatically gates the clock to the unused host controller or the unused peripheral controller.

This function is enabled if the NONUSE_CLK_MSK bit (bit 31) is set to 1b.

If it is no problem whether the clock supply to the unselected function is stopped, use this function.

The following table shows the operating specifications of this function, based on the setting of the OTG_PERI bit (bit 31) of the Common Control Register (offset: H’800).

Clock Gating Register Setting			Host/Peripheral Switching Setting	Gating Target	
			Register	Host Controller	Function Controller
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI		
1	0	0	0		✓
			1	✓	

[Note on using the NONUSE_CLK_MSK bit]

If the NONUSE_CLK_MSK bit is used, as the general rule, set the HOST_CLK_MSK/PERI_CLK_MSK bit to 0b.

If the HOST_CLK_MSK/PERI_CLK_MSK bit is 1b, the effect of clock gating becomes logical OR of each bit.

32A.3.1.3 Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations

This function forcibly gates the clock to the host controller and function controller.

The following table shows the operating specifications of this function.

Clock Gating Register Setting			Host/Peripheral Switching Setting	Gating Target	
			Register		
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI	Host Controller	Function Controller
0	1	0	—	✓	
	0	1	—		✓
	1	1	0	✓	✓

Note: If the HOST_CLK_MSK/PERI_CLK_MSK bit is used, as the general rule, set the NONUSE_CLK_MSK bit to 0b.
If the NONUSE_CLK_MSK bit is set t0 1b, the effect of clock gating becomes logical OR of each bit.

32A.4 Interrupt Sources

32A.4.1 Interrupt Signals

This module has the five interrupt signals listed below.

Interrupt Source Name	Interrupt Type	Pulse/ Level	Active Level
U2H_INT	BUS Master interrupt signal. This signal is asserted when a bus error occurs in the BUS Master. Interrupt control is performed by the AHB Bridge Register.	Level	H
U2H_OHCI_INT	OHCI interrupt signal This signal is asserted during FS/LS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the OHCI Operational Register.	Level	H
U2H_EHCI_INT	EHCI interrupt signal. This signal is asserted during HS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_WAKEON_INT	EHCI Wakeup interrupt signal. This signal is asserted by an EHCI Wakeup event. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_OBINT	OTG/Battery Charging interrupt signal. This signal is asserted by OTG or Battery Charging related event. Interrupt control is performed by the UCOM2 Register.	Level	H

32A.4.2 Interrupt Sources and Control

32A.4.2.1 U2H_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the register below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H’200) bit[0] (AHB_INTEN)

[Interrupt source]

A bus error (MHRESP = 1b) occurs in the AHB master.

[Clearing the interrupt]

To clear the interrupt, write 1b to the relevant bit in the register below.

AHB Bridge Register INT_STATUS Register (offset:H’204) bit[0] (AHB_INT)

32A.4.2.2 U2H_OHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H'200) bit[1] (USBH_INTAEN)

OHCI Operational Register HcInterruptEnable Register (offset:H'010) bit[31], bit[6:0]*[1]

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting							
		USBH_INTAEN	HcInterruptEnable						
			bit [31] MIE	bit [6] RHSCE	bit [5] FNOE	bit [3] ROE	bit [2] SFE	bit [1] WDHE	bit [0] SOE
1	Device connection is detected.	✓	✓	✓					
2	Device disconnection is detected.	✓	✓	✓					
3	Port power is OFF (excluding overcurrent detection).	✓	✓	✓					
4	Babble error is detected during a USB transfer.	✓	✓	✓					
5	Resume is complete.	✓	✓	✓					
6	Overcurrent is detected.	✓	✓	✓					
7	Bus reset is complete.	✓	✓	✓					
8	When the HcRhDescritorB Register DR bit is 1b, OHCI becomes "USB Operational" (HCFS[1:0] bit = 10b) or "USB Suspend" (HCFS[1:0] bit = 11b).	✓	✓	✓					
9	When no device is connected (CCS bit = 0b), 1 is written to bit [0] (Clear Port Enable) of the OHCI HcRhPort Status register.	✓	✓	✓					
10	When no device is connected (CCS bit = 0b), 1 is written to bit [1] (Set Port Enable) of the OHCI HcRhPort Status register.	✓	✓	✓					
11	When no device is connected (CCS bit = 0b), 1 is written to bit [2] (Set Port Suspend) of the OHCI HcRhPort Status register.	✓	✓	✓					
12	When no device is connected (CCS bit = 0b), 1 is written to bit [3] (Clear Suspend Status) of the OHCI HcRhPort Status register.	✓	✓	✓					
13	When no device is connected (CCS bit = 0b), 1 is written to bit [4] (Set Port Reset) of the OHCI HcRhPort Status register.	✓	✓	✓					
14	When no device is connected (CCS bit = 1b), the port power is turned off.	✓	✓	✓					
15	The MSB of bit [15:0] (Frame Number) of the HcFmNumber register has changed.	✓	✓		✓				
16	RemoteWakeup signal (Resume signal) is detected from a device.	✓	✓			✓			
17	HccaFrameNumber is updated. (Almost the same meaning as SOF is sent.)	✓	✓				✓		
18	A transfer finishes (including an error), and the host module updated HccaDoneHead.	✓	✓					✓	
19	USB schedule overrun occurred for the frame.	✓	✓						✓

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

OHCI Operational Register HcInterruptStatus Register (offset:H'00C) bit[6:0]

32A.4.2.3 U2H_EHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.
AHB Bridge Register INT_ENABLE Register (offset:H’200) bit[2] (USBH_INTBEN)

EHCI Operational Register USBINTR Register (offset:H’128) bit[16], bit[5:0]*[1]

Note 1. Enable the bit required as the assertion source.

Also, control the relevant bit(s) in the register below as required.

EHCI Operational Register USBCMD Register (offset:H’120) bit[15], bit[6]

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting								
			USBCMD		USBINTR					
			bit [15]	bit [6]	bit [16]	bit [5]	bit [3]	bit [2]	bit [1]	bit [0]
		USBH_INTB EN	Per-Port Change Event	Doorbell	Port-1 Change Event	Async Advance	Frame List Rollover	Port Change Detect	USB ERRINT	USB INT
[1]	Device connection is detected.	✓						✓		
[2]	Device disconnection is detected.	✓						✓		
[3]	Overcurrent is detected.	✓						✓		
[4]	RemoteWakeup signal (Resume Signal) is detected from a device.	✓								
[5]	Babble status of the USB bus is detected.	✓						✓		
[6]	USB transfer with “qTD IOC = 1b” normally finishes.	✓								✓
[7]	Short packet is received.	✓								✓
[8]	USB transfer finished with an error. (Retry transfer failed three times. A bubble error was detected. STALL was received.)	✓							✓	
[9]	The QH processing normally finished while the USBCMD Register bit [6] (Interrupt on Async Advance Doorbell) is 1b.	✓		✓		✓				
[10]	The FRINDEX Register FrameIndex bit returned from the maximum value to H’000 (rollover detected).	✓					✓			
[11]	The Port Change Detect event (interrupt source 1 to 5) was detected.	✓	✓		✓					

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

EHCI Operational Register USBSTS Register (offset:H’124) bit[16], bit[5:0]

32A.4.2.4 U2H_WAKEON_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H’200) bit[4] (WAKEON_INTEN)

EHCI Operational Register PORTSC[1:2] Register (offset:H’164/H’168) bit[22:20]*[1]

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting			
			PORTSC		
		WAKEON_INTEN	bit [22]	bit [21]	bit [20]
			WKOC_E	WKDSCNNT_E	WKCNNT_E
1	Device connection is detected.	✓			✓
2	Device disconnection is detected.	✓		✓	
3	Overcurrent is detected.	✓	✓		
4	RemoteWakeup signal (Resume Signal) is detected from a device.	✓			

[Clearing the interrupt]

To clear the interrupt, write 1b to the relevant bit in the register below.

AHB Bridge Register INT_STATUS Register (offset:H’204) bit[4] (WAKEON_INT)

32A.4.2.5 U2H_OBINT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset:H’200) bit[3] (UCOM_INTEN)

UCOM Register OTG-BC Interrupt Enable Register (offset:H’808) bit[17:16], bit[6], bit[4], bit[1:0]*[1]

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [17]	bit [16]
		UCOM_INTEN	DPMONCHG_EN	DMMONCHG_EN
1	The DP pin has changed.	✓	✓	
2	The DM pin has changed.	✓		✓

Interrupt Source		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [6]	bit [4]
		UCOM_INTEN	CHGDETCHG1_EN	PDDETCHG1_EN
1	The portable device detection signal has changed.	✓		✓
2	The charging port detection signal has changed.	✓	✓	

Interrupt Source		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [1]	bit [0]
		UCOM_INTEN	OCINT_EN	IDCHG_EN
1	Overcurrent is detected. (The change of the OVRCUR pin from 1 to 0b is detected.)	✓	✓	
2	The OTG_ID pin has changed.	✓		✓

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

UCOM Register OTG-BC Interrupt Status Register (offset:H’804) bit[17:16], bit[6], bit[4], bit[1:0]

32A.4.3 Timing of De-asserting Interrupt Signals

After the register access to clear an interrupt source, it may take time to begin clearing the interrupt triggered by the interrupt source. Therefore, take a measure to prevent the false recognition of interrupts during the period from the end of the register clear access until the next interrupt is recognized.

32A.5 Power-Saving Function

This module controls power consumption by using the following two methods:

1. Controlling the SUSPENDM/SLEEPM pin of USBPHY
2. Using clock gating to stop the clock to the host controller or peripheral controller.

32A.5.1 Controlling the SUSPENDM and SLEEPM Pins of the USBPHY

You can expect the following power-saving effects by asserting the SUSPENDM and SLEEPM pins of the USBPHY:

- Reducing the power consumption by the USBPHY
- Reducing the power consumption by the host core by stopping the clocks from the USBPHY

As described at **Section 32A.1.2.5, Suspend extension function**, by default, the SUSPENDM and SLEEPM pins of the USBPHY are not asserted even when EHCI and OHCI are put into the Suspended state.

See **Section 32A.2.4.4(6), Suspend Control Register** and control the relevant registers appropriately.

32A.5.2 Controlling the Clock-Gating Function

See **Section 32A.2.4.4(5), Register Enable/Clock Gating Control Register** and **Section 32A.3.1, Clock Gating Specifications**, then control the relevant registers appropriately.

32A.6 Battery Charging

32A.6.1 Support of charging port

The charging port refers to a port that supplies power in compliance with the Battery Charging Specification. The charging port is usually installed on the host controller.

Charging ports can be generally classified by function as described below.

Type	Function
CDP (Charging Downstream Port)	Downstream port that can supply power in compliance with the Battery Charging Specification. This type of charging port detects a portable device, and after the handshake (for Battery Charging) finishes, it proceeds to the usual device connect sequence (and operates as the usual host).
DCP (Dedicated Charging Port)	Port that provides only power supply function in compliance with the Battery Charging Specification. This type of charging port does not operate as the usual host.
SDP (Standard Downstream Port)	Standard Downstream Port that is not compliant with the Battery Charging Specification. This type of charging port supplies power in the range conforming to the conventional USB 2.0 standard, and operates as the usual host.

32A.6.2 Support of portable device

A portable device refers to a device that is supplied power (or that requests power supply) in compliance with the Battery Charging Specification. The portable port is usually installed in the peripheral controller.

32A.7 Bus Master

32A.7.1 Functional specifications of the bus master

32A.7.1.1 Supported bus master functions

Function as the BUS MASTER	Status
Residual burst after an error response is received	The transfer is not stopped.
1 Kbyte boundary processing	The fixed length burst (INCRx) across 1 Kbyte boundary is not performed.

32A.7.1.2 Issuing requests for different types of bus transfer

MHTRANS[1:0]	MHSIZE[2:0]	MHWRITE	MHBURST[2:0]	Reply	Remarks
IDLE (00b)	—	—	—	—	—
BUSY (01b)	—	—	—	—	Not issued.
NONSEQ (10b)	32-bit (010b)	WRITE	SINGLE INCR4 INCR8 INCR16	OKAY/ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ			
	8-bit (000b) 16-bit (001b)	WRITE	SINGLE	OKAY/ERROR	8-bit/16-bit transfer is issued (SINGLE only). A response error is reported by an interrupt.
	other than the above	—	—	—	Not issued.
SEQ (11b)	32-bit (010b)	WRITE	INCR4 INCR8 INCR16	OKAY/ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ		OKAY/ERROR	
	other than the above	—	—	—	Not issued.

32A.7.1.3 Supported responses

Response type	Response	Remarks
OKAY	Enable	Supported.
ERROR	Enable	A response error is reported by an interrupt, and termination of transfer (Early Burst Termination) is not performed.

32A.7.1.4 Protection control information

The value of MHPROT[3:0] can be set in the PROT_TYPE bits (bits [15:12]) in the AHB_BUS_CTR Register (offset: H’208).

Also, when the PROT_MODE bit (bit 8) in the AHB_BUS_CTR Register is set, only the last data transfer in an EHCI/ OCHI DMA transfer can be handled as a non-buffered transfer, and other transfers can be handled as buffered transfers.

32A.7.1.5 Maximum burst length

The maximum burst length can be selected from SINGLE, INCR4, INCR8, and INCR16 by using the MAX_BURST_LEN bits (bit[1:0]) in the AHB_BUS_CTR Register (offset: H’208). The maximum burst length is common to reading and writing.

32A.7.1.6 Boundary of transfer data

The value of MHADDR[31:0] bits does not exceed 1 KB boundary during a burst transfer. Also, you can change the address boundary for burst transfer to 16, 32, or 64 bytes by writing a value to the ALIGN_ADDRESS bits (bit[5:4]) in the AHB_BUS_CTR Register (offset: H’208).

32A.7.1.7 Start address of fixed-length INCR burst transfer

The following table lists the values of the lower bits of MHADDR to be applied when a fixed-length INCR burst transfer starts.

ALIGN_ADDRESS Setting	Fixed length Start Address of INCR Burst		
	INCR4	INCR8	INCR16
00b (Aligned at the 1-Kbyte boundary)	MHADDR[9:0] = H’000 H’004 H’008 H’00C H’010 : H’3D0 H’3D4 H’3D8 H’3DC H’3E0 H’3E4 H’3E8 H’3EC H’3F0	MHADDR[9:0] = H’000 H’004 H’008 H’00C H’010 : H’3C8 H’3CC H’3D0 H’3D4 H’3D8 H’3DC H’3E0	MHADDR[9:0] = H’000 H’004 H’008 H’00C H’010 : H’3B8 H’3BC H’3C0
01b (Aligned at the 16-byte boundary)	MHADDR[3:0] = H’0	-- (Not issued)	-- (Not issued)
10b (Aligned at the 32-byte boundary)	MHADDR[4:0] = H’00 H’04 H’08 H’0C H’10	MHADDR[4:0] = H’00	-- (Not issued)
11b (Aligned at the 64-byte boundary)	MHADDR[5:0] = H’00 H’04 H’08 H’0C H’10 H’14 H’18 H’1C H’20 H’24 H’28 H’2C H’30	MHADDR[5:0] = H’00 H’04 H’08 H’0C H’10 H’14 H’18 H’1C H’20	MHADDR[5:0] = H’00

32A.8 Overcurrent Control and VBUS Control

32A.8.1 OVRCUR/VBUSEN pin

Overcurrent detection on the USB port and port power (VBUS) control are performed by the external power IC connected to this module.

This module pin	Input/Output	Level	Description
OVRCURI	Input	L	Overcurrent status was detected.
		H	Overcurrent status was not detected.
VBUSEN	Output	L	Port Power (VBUS) OFF
		H	Port Power (VBUS) ON

32A.8.2 Overcurrent detection timer setting

This module detects overcurrent when the OVRCUR pin remains asserted (0b) for a set period.

The period over which assertion of the OVRCUR pin is required (“overcurrent detection time”) can be set in the following register.

Register	Overcurrent detection/sleep timer setting register (offset: H’310)
Bits	TIMER_OC[19:0]
Initial value	H’3_0D40

The overcurrent detection time can be converted from the setting value of the above register, taking that one bit equal to the internal bus clock (P1ϕ) cycle.

Therefore, set the value of the above register at initial configuration, considering “the internal bus clock (P1ϕ) frequency to be used” and “the overcurrent detection time you want to specify”.

32A.8.3 Port Power (VBUS) control specifications

VBUSEN can be controlled by the Port Power bit of the EHCI/OHCI Operational register or by the VBOUT bit of the VBUS Control Register. Which of the above bit controls VBUSEN is determined by the VBUSEN control (VBENCTL) register in USBPHY Control.

Refer to **Section 32.3.4, VBUSEN control register (VBENCTL)**.

When using as Function, VBUSEN should be 0.

Situation		Register		bit
EHCI control		PORTSC1 (offset: H’164)		bit 12 (PP)
OHCI control	Global control*1	ON setting	HcRhStatus Register (offset: H’050)	bit 16 (Set Global Power)
		OFF setting		bit 0 (Clear Port Status)
	Selective control*2	ON setting	HcRhPortStatus1 Register (offset: H’054)	bit 8 (Set Port Power)
		OFF setting		bit 9 (Clear Port Power)

Note 1. Global control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: H’048) bit 8 (PSM) = 0b
or
HcRhDescriptorA Register (offset: H’048) bit 8 (PSM) = 1b
and
HcRhDescriptorB Register (offset: H’04C) bit 17 (PPCM[1]) = 0b

Note 2. Selective control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: H’048) bit 8 (PSM) = 1b
and
HcRhDescriptorB Register (offset: H’04C) bit 17 (PPCM[1]) = 1b

However, if the register settings are as follows, the VBUSEN pin is always asserted (1b), and the Port Power (VBUS) becomes ON, regardless of the OVRCUR pin’s status.

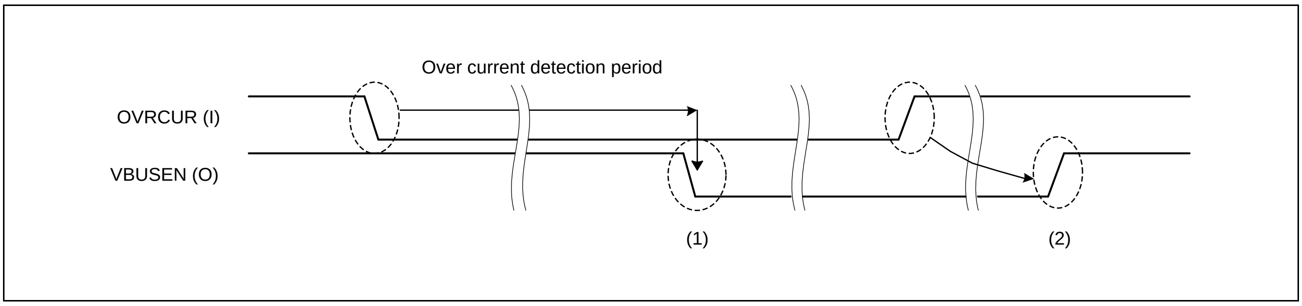
Specify the settings as necessary (for example, your system needs that VBUS is always ON).

EHCI Operational	OHCI Operational				
HCSPARAMS (offset: H’104)	HcRhDescriptorA (offset: H’048)			HcRhDescriptorB (offset: H’04C)	Pin Operation
PPC (bit 4)	NOCP (bit 12)	NPS (bit 9)	PSM (bit 8)	PPCM[1] (bit 17)	When the OVRCUR pin is asserted (0b)
0	—	—	—	—	Fixed to 1b
—	1	—	—	—	Fixed to 1b
—	—	1	—	—	Fixed to 1b
1	0	0	0	—	0b
			1	0	
				1	

32A.8.4 Timing Chart for Overcurrent Detection and Recovery

The figure below shows the assertion/de-assertion timings of the OVRCUR and VBUSEN pins signals at overcurrent detection and recovery.

Note that this timing chart is on the assumption that the changes of register settings to fix the Port Power bit to asserted state (see **Section 32A.8.3, Port Power (VBUS) control specifications**) have not been made.



1. When the OVRCUR pin is kept asserted (0b) for the overcurrent detection time, this module determines the occurrence of overcurrent, and then de-asserts the VBUSEN pin (0b).
2. After the overcurrent status has been resolved, and de-assertion of the OVRCUR pin (1b) is confirmed, 1b is written to the Port Power bit described in **Section 32A.8.3, Port Power (VBUS) control specifications** to turn on the Port Power (Vbus).

NOTE

Before the Port Power bit is set by firmware, be sure to check that the OVRCUR pin has been deasserted.

32A.9 Procedure for Setting this Module

32A.9.1 Host/Peripheral Common Setting Sequence

The following shows the necessary sequence common to both host and peripheral modes.

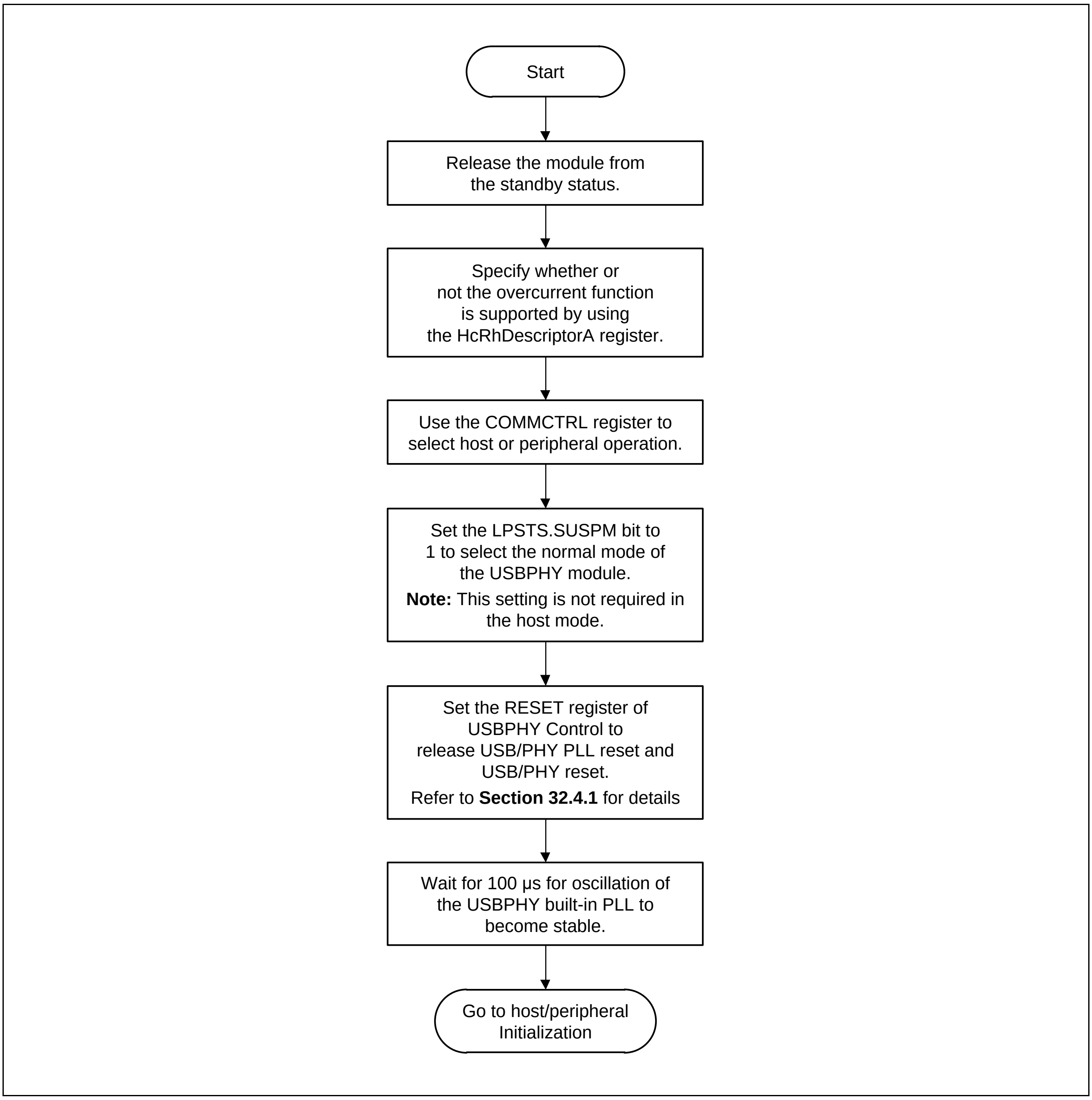


Figure 32A.1 Sequence Common to Both Host and Peripheral Modes

32A.9.2 Initialization Sequence

The following shows the initialization sequence in the Host mode

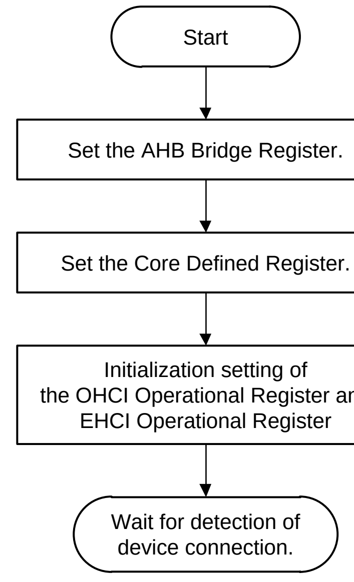


Figure 32A.2 Initialization Sequence

32A.9.3 Flow of Error Handling

While operating this module, if the operation falls into an abnormal state and recovery to the normal flow seems difficult, perform the following steps for reset.

[When an abnormality occurs while EHCI is running]

1. Write 1b to the HCRESET bit (bit 1) of the EHCI Operational Register USBCMD Register (offset: H’120) to execute EHCI software reset.
2. Re-initialize the EHCI Operational Register.

[When an abnormality occurs while OHCI is running]

1. Write 1b to the HCR bit (bit 0) of the OHCI Operational Register HcCommandStatus Register (offset: H’008) to execute OHCI software reset.
2. Re-initialize the OHCI Operational Register.

32B. USB 2.0 Function Module

32B.1 Overview

32B.1.1 Overview

This LSI has two channels of USB 2.0 host/function module. Switching between the host and peripheral functions for each channel is possible by setting the UCOM register.

The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

This chapter describes the peripheral controller. For details on the host/peripheral common circuit and battery charging, see **Section 32A, USB 2.0 Host Module**.

This module is a universal serial bus (USB) controller that has peripheral functions.

This module supports high-speed and full-speed transfer defined by the Universal Serial Bus Specification Revision 2.0.

This module supports all transfer types defined in the USB Specification. This module incorporates 8 Kbytes of buffer memory for data transfer, and can use a maximum of 10 pipes. You can assign any endpoint number to any pipe other than pipe 0 in conformity to the peripheral equipment or system to communicate with.

32B.1.2 Features

32B.1.2.1 Peripheral controller supporting high-speed USB

(1) On-chip peripheral USB controller

32B.1.2.2 Support of all types of USB transfer

(1) Supporting all types of USB transfer, including isochronous transfer

(2) Control transfer

(3) Bulk transfer

(4) Interrupt transfer (high-bandwidth transfers not supported)

(5) Isochronous transfer (high-bandwidth transfers not supported)

32B.1.2.3 Bus interface

(1) Includes a two-channel DMA interface

32B.1.2.4 Pipe configuration

(1) 8 Kbytes of buffer memory for USB communications for each channel

(2) Up to 10 pipes (including the default control pipe) selectable for each channel

(3) Programmable pipe configuration

(4) Any endpoint number assignable to pipes other than pipe 0

Table 32B.1 Pipe Settings

PIPE	Transfer Type	Double buffer	Continuous transfer mode	Buffer size
PIPE0	Control	—	—	Fixed to 64 bytes/ 256 bytes (CNTMD = 1)
PIPE1	Iso/Bulk	✓	✓ (Bulk only)	Up to 2 Kbytes
PIPE2	Iso/Bulk	✓	✓ (Bulk only)	Up to 2 Kbytes
PIPE3	Bulk	✓	✓	Up to 2 Kbytes
PIPE4	Bulk	✓	✓	Up to 2 Kbytes
PIPE5	Bulk	✓	✓	Up to 2 Kbytes
PIPE6	Int	—	—	Fixed to 64 bytes
PIPE7	Int	—	—	Fixed to 64 bytes
PIPE8	Int	—	—	Fixed to 64 bytes
PIPE9	Int	—	—	Fixed to 64 bytes

32B.1.2.5 Features of peripheral functions

(1) Support of high-speed transfer (at 480 Mbps) and full-speed transfer (at 12 Mbps)

(2) Automatic recognition of high-speed or full-speed operation based on automatic response to the reset handshake

(3) Control transfer stage monitoring function

(4) Device state monitoring function

(5) Automatic response to SET_ADDRESS request

(6) NAK response interrupt (NRDY)

(7) SOF interpolation

32B.1.2.6 Features of DMA transfer

DMA transaction mode:	Fetching in both register and link modes are supported.
Interrupt:	Level is supported.
Transfer size:	A transfer size from 1 to 128 bytes can be selected separately for the transfer source and transfer destination.
Skip (scatter/gather) function:	The access size and skip size can be specified separately for the transfer source and destination.
Suspend function:	A running DMA transaction can be suspended temporarily.
Interval function:	The interval of DMA transfers can be specified to control the bus occupancy.

32B.1.2.7 Other functions

(1) Byte endian swap function to support both big endian and little endian as data formats (when using only CFIFO)

(2) Transfer ending function using a transaction counter

(3) SOF pulse output function

(4) BRDY interrupt event notification timing change function (BFRE)

(5) Function (SHTNAK) to set NAK in the response PID when transfer ends

(6) Support of the Link Power Management (LPM) ECN, making available a new low-power-consumption state (L1 state)

32B.1.3 Overview of Functions

32B.1.3.1 Automatic recognition of USB transfer speed

This module automatically recognizes USB transfer speed.

(1) Methods of FIFO buffer memory access

This module supports the two types of access described below to the FIFO buffer memory for USB data transfer.

(a) Access from the CPU

Specify a FIFO port address, and then write data to or read data from the FIFO buffer memory.

(b) Direct memory access (DMA)

Selecting a pipe window and setting the DMA control registers enables writing data to or reading data from the FIFO buffer memory.

32B.1.3.2 USB event

This module notifies the event in USB operation by issuing an interrupt.

You can specify whether to enable notification by interrupt for individual interrupt types and sources through software settings.

32B.1.3.3 USB data transfer

This module performs all types of USB data transfer: control transfer, bulk transfer, interrupt transfer, and isochronous transfer. The following pipe resources are available for individual transfer types:

(1) One pipe dedicated to control transfer

(2) Four pipes dedicated to interrupt transfer

(3) Three pipes dedicated to bulk transfer

(4) Two pipes selectively used for bulk or isochronous transfer

For each pipe, specify the settings, including transfer type, endpoint number, and maximum packet size, required for USB transfer according to the system.

This module can incorporate up to 8 Kbytes of buffer memory. For the pipes dedicated to bulk transfer and those selectively used for bulk transfer or isochronous transfer, allocate buffer memory and specify a buffer operating mode and other necessary settings according to the system. Setting the buffer operating mode enables high-speed data transfers with fewer interrupts to be performed by using double-buffering and continuous transfer of data packets.

32B.1.3.4 SOF pulse output function

This module has a function to output an SOF pulse to indicate the timing of SOF packet transmission. This module asserts a SOF pulse output signal when an SOF packet is received. This module outputs pulses at regular intervals based on an SOF interpolation timer even when an SOF packet is damaged.

32B.1.4 Restriction matter and Notes

32B.1.4.1 Restriction matter

(1) Restrictions on the USB specifications

It’s no support About below of USB 2.0 specification

- HighBandWidth transfer is no support

(2) Restrictions on DMA Master

- DAD = 1 (destination address fixed) and can not use skip transfer of destination side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- SAD = 1 (source address fixed), and can not use skip transfer of source side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- DAD = 1 (destination address fixed), and can not use the beat unalign transfer of destination side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- SAD = 1 (source address fixed), and can not use beat unalign transfer of source side. When forwarding by such setting, movement is unsettled. Should not do such transmission.
- When REQD =1, SBE = 1 (sweep mode) and compulsion discharge function can not use.

32B.1.4.2 Notes

(1) DMA transfer and passing problem of interrupt signal in DMA Master construction.

(a) Overview

A USBFDMAmn interrupt (m, n = 0, 1) might occur before the last data of DMA transaction is written to the write-target device.

(b) Plan to avoid

A plan to avoid the above problem is indicated below.

Plan to avoid 1) HPROT is set as non bufferable.

According to the DMA mode, set 0 in the DPR[2] bit in the CHEXT_n register or the LDPR[2] bit in the DCTRL register, set the HPROT signal as non bufferable, and then perform the DMA transaction.

If all transfers are set to non bufferable, transfer efficiency might fall.

In such cases, set the most part of transaction as bufferable and perform a transfer, and then perform the last transfer by setting a register set or descriptor as non bufferable.

(2) Setting for battery charging when the peripheral controller is selected

The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

32B.2 Registers

How to read the register table

(1) Bit number:

(2) State after reset: Initial state of the register that occurs immediately after a reset "Power on Reset" indicates the initial state at power-on reset.
The state after USB reset is the initial state of the register that occurs when this module detects a USB bus reset.
Significant points regarding reset operation are indicated in notes.
"—" indicates that a user's setting is retained without this module operation having been performed. "X" indicates that the value is undefined.

(4) Access condition: The condition to be met when this module accesses the register for an operation.

R: Reading only
W: Writing only
R/W: Reading or writing
R(0): 0-reading only
W(1): 1-writing only

(5) Name: Bit symbol and bit name

(6) Function: Description of functions.

<Example of description>

(1) Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	A bit	B bit	C bit	—	—	—	—	—	—	—	—	—	—	—	—
(2) Initial Value	X	0	0	0												
	X	0	—	—												
R/W																

Bit	Bit Name	Initial Value	R/W	Description
15	—			Nothing is assigned. Fix this bit to 0.
14	A bit		R/W	AAA enable 0: Disables operation 1: Enables operation
13	B bit		R	BBB operation 0: Outputs low-level signal 1: Outputs high-level signal
12	C bit		R(0)/ W(1)	CCC control 0: .... 1: ....
	(5)		(4)	(6)

32B.2.1 Base Address

Table 32B.2 Base addresses for each channel of the USB function module

Channel number	Base address
0	H’0_11C6_0000 (Cortex-A55 Address Space) H’41C6_0000 (Cortex-M33 Address Space Non-Secure) H’51C6_0000 (Cortex-M33 Address Space Secure)

32B.2.2 List of Registers

Table 32B.3 lists the registers of this module.

Table 32B.3 List of Registers (1/3)

Offset Address	Symbol	Name	Access unit
000	SYSCFG0	System Configuration Control Register 0	16-bit
002	SYSCFG1	System Configuration Control Register 1	16-bit
004	SYSSTS0	System Configuration Status Register	16-bit
008	DVSTCTR0	Device Control Register 0	16-bit
00C	TESTMODE	Test Mode Register	16-bit
014	CFIFO	CFIFO Port Register	8/16/32-bit
016			
020	CFIFOSEL	CFIFO Port Select Register	16-bit
022	CFIFOCTR	CFIFO Port Control Register	16-bit
028	D0FIFOSEL	D0FIFO Port Select Register	16-bit
02A	D0FIFOCTR	D0FIFO Port Control Register	16-bit
02C	D1FIFOSEL	D1FIFO Port Select Register	16-bit
02E	D1FIFOCTR	D1FIFO Port Control Register	16-bit
030	INTENB0	Interrupt Enable Register 0	16-bit
036	BRDYENB	BRDY Interrupt Enable Register	16-bit
038	NRDYENB	NRDY Interrupt Enable Register	16-bit
03A	BEMPENB	BEMP Interrupt Enable Register	16-bit
03C	SOFCFG	SOF Output Configuration Register	16-bit
040	INTSTS0	Interrupt Status Register 0	16-bit
046	BRDYSTS	BRDY Interrupt Status Register	16-bit
048	NRDYSTS	NRDY Interrupt Status Register	16-bit
04A	BEMPSTS	BEMP Interrupt Status Register	16-bit
04C	FRMNUM	Frame Number Register	16-bit
04E	UFRMNUM	Micro Frame Number Register	16-bit
050	USBADDR	USB Address Register	16-bit
054	USBREQ	USB Request Type Register	16-bit
056	USBVAL	USB Request Value Register	16-bit
058	USBINDX	USB Request Index Register	16-bit
05A	USBLENG	USB Request Length Register	16-bit
05C	DCPCFG	DCP Configuration Register	16-bit
05E	DCPMAXP	DCP Max. Packet Size Register	16-bit

Table 32B.3 List of Registers (2/3)

Offset Address	Symbol	Name	Access unit
060	DCPCTR	DCP Control Register	16-bit
064	PIPESEL	Pipe Window Select Register	16-bit
068	PIPECFG	Pipe Configuration Register	16-bit
06A	PIPEBUF	Pipe Buffer Setting Register	16-bit
06C	PIPEMAXP	Pipe Max. Packet Size Register	16-bit
06E	PIPEPERI	Pipe Cycle Control Register	16-bit
070	PIPE1CTR	PIPE1 Control Register	16-bit
072	PIPE2CTR	PIPE2 Control Register	16-bit
074	PIPE3CTR	PIPE3 Control Register	16-bit
076	PIPE4CTR	PIPE4 Control Register	16-bit
078	PIPE5CTR	PIPE5 Control Register	16-bit
07A	PIPE6CTR	PIPE6 Control Register	16-bit
07C	PIPE7CTR	PIPE7 Control Register	16-bit
07E	PIPE8CTR	PIPE8 Control Register	16-bit
080	PIPE9CTR	PIPE9 Control Register	16-bit
090	PIPE1TRE	PIPE1 Transaction Counter Enable Register	16-bit
092	PIPE1TRN	PIPE1 Transaction Counter Register	16-bit
094	PIPE2TRE	PIPE2 Transaction Counter Enable Register	16-bit
096	PIPE2TRN	PIPE2 Transaction Counter Register	16-bit
098	PIPE3TRE	PIPE3 Transaction Counter Enable Register	16-bit
09A	PIPE3TRN	PIPE3 Transaction Counter Register	16-bit
09C	PIPE4TRE	PIPE4 Transaction Counter Enable Register	16-bit
09E	PIPE4TRN	PIPE4 Transaction Counter Register	16-bit
0A0	PIPE5TRE	PIPE5 Transaction Counter Enable Register	16-bit
0A2	PIPE5TRN	PIPE5 Transaction Counter Register	16-bit
0D0	DEVADD0	Device Address 0 Configuration Register	16-bit
0D2	DEVADD1	Device Address 1 Configuration Register	16-bit
0D4	DEVADD2	Device Address 2 Configuration Register	16-bit
0D6	DEVADD3	Device Address 3 Configuration Register	16-bit
0D8	DEVADD4	Device Address 4 Configuration Register	16-bit
0DA	DEVADD5	Device Address 5 Configuration Register	16-bit
0DC	DEVADD6	Device Address 6 Configuration Register	16-bit
0DE	DEVADD7	Device Address 7 Configuration Register	16-bit
0E0	DEVADD8	Device Address 8 Configuration Register	16-bit
0E2	DEVADD9	Device Address 9 Configuration Register	16-bit
0E4	DEVADDA	Device Address A Configuration Register	16-bit
100	LPCTRL	Low Power Control Register	16-bit
102	LPSTS	Low Power Status Register	16-bit
104	PHYFUNCTR	PHY Function Control Register	16-bit
10A	PHYOTGCTR	PHY OTG Control Register	16-bit
144	PL1CTRL1	Peripheral L1 Control Register 1	16-bit
146	PL1CTRL2	Peripheral L1 Control Register 2	16-bit
400	N0SA_0	Next0 Source Address Register 0	32-bit
404	N0DA_0	Next0 Destination Address Register 0	32-bit
408	N0TB_0	Next0 Transaction Byte Register 0	32-bit

Table 32B.3 List of Registers (3/3)

Offset Address	Symbol	Name	Access unit
40C	N1SA_0	Next1 Source Address Register 0	32-bit
410	N1DA_0	Next1 Destination Address Register 0	32-bit
414	N1TB_0	Next1 Transaction Byte Register 0	32-bit
418	CRSA_0	Current Source Address Register 0	32-bit
41C	CRDA_0	Current Destination Address Register 0	32-bit
420	CRTB_0	Current Transaction Byte Register 0	32-bit
424	CHSTAT_0	Channel Status Register 0	32-bit
428	CHCTRL_0	Channel Control Register 0	32-bit
42C	CHCFG_0	Channel Configuration Register 0	32-bit
430	CHITVL_0	Channel Interval Register 0	32-bit
434	CHEXT_0	Channel Extension Register 0	32-bit
438	NXLA_0	Next Link Address Register 0	32-bit
43C	CRLA_0	Current Link Address Register 0	32-bit
440	N0SA_1	Next0 Source Address Register 1	32-bit
444	N0DA_1	Next0 Destination Address Register 1	32-bit
448	N0TB_1	Next0 Transaction Byte Register 1	32-bit
44C	N1SA_1	Next1 Source Address Register 1	32-bit
450	N1DA_1	Next1 Destination Address Register 1	32-bit
454	N1TB_1	Next1 Transaction Byte Register 1	32-bit
458	CRSA_1	Current Source Address Register	32-bit
45C	CRDA_1	Current Destination Address Register 1	32-bit
460	CRTB_1	Current Transaction Byte Register 1	32-bit
464	CHSTAT_1	Channel Status Register 1	32-bit
468	CHCTRL_1	Channel Control Register 1	32-bit
46C	CHCFG_1	Channel Configuration Register 1	32-bit
470	CHITVL_1	Channel Interval Register 1	32-bit
474	CHEXT_1	Channel Extension Register 1	32-bit
478	NXLA_1	Next Link Address Register 1	32-bit
47C	CRLA_1	Current Link Address Register 1	32-bit
600	SCNT_0	Source Continuous Register 0	32-bit
604	SSKP_0	Source Skip Register 0	32-bit
608	DCNT_0	Destination Continuous Register 0	32-bit
60C	DSKP_0	Destination Skip Register 0	32-bit
620	SCNT_1	Source Continuous Register 1	32-bit
624	SSKP_1	Source Skip Register 1	32-bit
628	DCNT_1	Destination Continuous Register 1	32-bit
62C	DSKP_1	Destination Skip Register 1	32-bit
700	DCTRL	DMA Control Register	32-bit
704	DSCITVL	Descriptor Interval Register	32-bit
710	DST_EN	DMA Status EN Register	32-bit
714	DST_ER	DMA Status ER Register	32-bit
718	DST_END	DMA Status END Register	32-bit
71C	DST_TC	DMA Status TC Register	32-bit
720	DST_SUS	DMA Status SUS Register	32-bit

32B.2.3 System Configuration Control Registers

32B.2.3.1 System Configuration Control Register 0 [SYSCFG0] <Address: H'000>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CNEN	HSE	—	DRPD	DPRPU	—	—	—	USBE
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
8	CNEN	0	R/W	This bit prohibits or enables single-ended receiver operation. 0: Single-ended receiver operation prohibited 1: Single-ended receiver operation enabled
7	HSE	0	R/W	This bit prohibits or enables High-Speed operation. 0: High-Speed operation prohibited (Full-Speed) 1: High-Speed operation enable (The controller detects the communication speed.)
6	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
5	DRPD	1	R/W	D+/D- line resistor control Set this bit to 0 to use this module. For details, see Control of USB Data Bus Resistors.
4	DPRPU	0	R/W	D+ line resistor control This bit prohibits or enables D+ line pull-up for the peripheral controller function. For details, see Control of USB Data Bus Resistors. 0: Pull Up prohibited 1: Pull Up enabled
3 to 1	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
0	USBE	0	R/W	USB block operation prohibited This bit prohibits or enables USB block operation. 0: USB block operation prohibited 1: USB block operation enabled

Note: Data can be written to and read from this register even while the UTMI+PHY clock is stopped.
However, if a value is set while the UTMI+PHY clock is stopped, the corresponding function takes effect after the oscillation of UTMI+PHY clock starts.

(1) Single-ended receiver operation enable (CNEN) bit

Setting this bit enables the single-ended receiver to operate. This bit is intended to prevent damage by inrush current that can be caused when the single-ended receiver in unattached status goes floating. This bit also allows the LNST bit to be referenced.

Set this bit when VBUS is detected as the result of a VBUS interrupt. Clear this bit when VBUS is removed.

(2) High-speed operation enable (HSE) bit

Setting this bit enables the high-speed operation. When this bit is 1, this module performs a high-speed or full-speed operation according to the result of reset handshake.

- When the HSE bit is 0, this module performs a full-speed operation.
- When the HSE bit is 1, this module executes the reset handshake protocol, and then automatically performs a high-speed or full-speed operation according to the result of reset handshake.

Rewriting the value of this bit must be done when the DPRPU bit is 0.

(3) D+/D- line resistor control (DRPD or DPRPU) bit

Table 32B.4 shows available settings of the resistors for the USB data bus. Use the DPRPU bit to select the USB data bus resistors.

Table 32B.4 Control of USB Data Bus Resistors

Setting		Control of USB Data Bus Resistors		
DRPD	DPRPU	D- Line	D+ Line	Remarks
0	0	Open	Open	
0	1	Open	Pull-Up	Specify the settings as shown in the left.
1	0	Pull-Down	Pull-Down	Initial state (When power on reset is canceled)
1	1	Pull-Down	Pull-Up	Setting prohibited

(a) D+ pull-up resistor control (DPRPU) bit

Setting this bit enables this module to notify the USB host of attaching by pulling up the D+ line voltage to 3.3 V. Clearing this bit enables this module to let the USB host know that the device has been detached by stopping pulling up the D+ line voltage.

(4) USB block operation enable (USBE) bit

This bit enables or disables the operation of the USB block of this module.

When this bit is changed from 1 to 0, this module initializes the bits shown in **Table 32B.5**.

Table 32B.5 Register Bits That Are Initialized by Writing 0 to the USBE Bit

Register Name	Bit Name
SYSSTS0	LNST
DVSTCTR0	RHST
INTSTS0	DVSQ
USBADDR	USBADDR
USBREQ	bRequest bmRequestType
USBVAL	wValue
USBINDX	wIndex
USBLENG	wLength

Note: Changing the value of this bit must be done when the SUSPENDM bit is 1 and after the oscillation of UTMI+PHY clock starts.

32B.2.3.2 System Configuration Control Register 1 [SYSCFG1] <Address: H’002>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	BWAIT					
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11 to 8	—	H’F	R	Nothing is assigned to these bits. Fix these bits to 0.
7, 6	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
5 to 0	BWAIT	H’0F	R/W	CPU bus access wait specification These bits specify the number of wait cycles for the access to this module. 000000: 0 wait cycles (2 access cycles) : 000010: 2 wait cycles (4 access cycles) : 000100: 4 wait cycles (6 access cycles) : 001111: 15 wait cycles (17 access cycles) (default) : 111111: 63 wait cycles (65 access cycles)

(1) CPU bus access wait specification (BWAIT) bits

These bits specify the wait cycles for the access to the HPB.

The following restriction is placed on the cycle of the access to the registers at address H’04 or after of this module:
Restriction on wait cycle: The cycle of continuous accesses to registers of this module must be at least 67 ns.

To comply with this restriction, you must control the number of wait cycles with the internal bus clock (P1ϕ) frequency.

The default of wait cycles is 17 clock cycles (maximum limit). Select an optimum setting.

This setting is also applied to accesses to FIFO port registers. The maximum speeds of accesses to FIFO ports are as follows:

MBW = 10 (32-bit access width): Max 60 MBytes/sec
MBW = 01 (16-bit access width): Max 30 MBytes/sec
MBW = 00 (8-bit access width): Max 15 MBytes/sec

32B.2.4 System Configuration Status

32B.2.4.1 System Configuration Status Register (SYSSTS0) <Address: H’004>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LNST	
Initial Value	—	—	0	0	0	0	0	0	0	0	0	0	0	—	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	See above	R	Nothing is assigned to these bits. Fix these bits to 0.
1, 0	LNST	All 0	R	USB Line status monitor The USB line status is displayed. *Note*: See the detailed description.

(1) Line status monitor (LNST) bits

Table 32B.6 shows the line status of the USB data bus of this module. This module monitors the line status (status of the D+ and D- lines) of the USB data bus in the LNST bits of the SYSSTS0 register.

Referencing the LNST bits must be done only after the USBE bit is set and attaching is performed (the DPRPU bit is set).

Table 32B.6 Line Status of USB Data Bus

LNST [1]	LNST [0]	Full-Speed operation	High-Speed operation	Chirp operation
0	0	SE0	Squelch	Squelch
0	1	J State	Unsquelch	Chirp J
1	0	K State	Invalid	Chirp K
1	1	SE1	Invalid	Invalid

Note: Chirp: State in which high-speed operation is enabled (HSE = 1) and the reset handshake protocol is being executed
Squelch: SE0 or idle state
Unsquelch: High-speed J or high-speed K state
Chirp J: Chirp J State
Chirp K: Chirp K State

32B.2.5 USB Signal Control Registers

32B.2.5.1 Device State Control Register 0 [DVSTCTR0] <Address: H'008>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WKUP	—	—	—	—	—	RHST		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R/W(1)	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R/W	Reserved. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
14 to 9	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
8	WKUP	0	R/W (1)	Remote wakeup output This bit prohibits or enables Remote wakeup (resume signal output). 0: Remote wakeup signal is not output. 1: Remote wakeup signal is output.
7 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2 to 0	RHST	All 0	R	Reset handshake This bit indicates the reset handshake status. *Note*: See the detailed description.

(1) Remote wakeup (resume signal output) enable (WKUP) bit

When this bit is set, this module outputs the remote wakeup signal to the USB.

This module manages the time of remote wakeup signal output. When the WKUP bit is set, this module outputs the K state for 10 ms, and then clears the WKUP bit.

The USB Specification requires the USB idle state to be retained for at least 5 ms before the remote wakeup signal is sent. Therefore, even if the WKUP bit is set immediately after the suspended state is detected, this module waits for 2 ms, and then outputs the K state.

Writing 1 to the WKUP bit must be done only when the device is in the suspended state (DVSQ = 1xx) and remote wakeup is allowed by the USB host.

When setting the WKUP bit, do not stop the internal clock even if the device is in the suspended state. (Write 1 to the WKUP bit when the SUSPM bit is 1.)

When the WKUP bit is set at a transition to the L1 state, this module outputs the K state for 50 µs, and then clears the WKUP bit. In the L1 state, setting the WKUP bit must be done only when the DVSQ[4] bit is 1.

(2) Reset handshake status (RHST) bits

This module outputs the result of reset handshake to this bit. **Table 32B.7** lists the results of reset handshake.

Table 32B.7 Reset Handshake Status

Bus State	Value of RHST Bit
Powered or disconnected state	000
Reset handshake in process	100
Full-speed connection	010
High-speed connection	011

If the HSE bit is 1, the RHST bits indicate 100 when this module detects a USB bus reset. Then, after this module has output Chirp K, these bits indicate 011 when this module detects Chirp JK from the USB host three times. If the status is not fixed to High-Speed within 2.5 ms after Chirp K is output, these bits indicate 010.

If the HSE bit is 0, the RHST bits indicate 010 when this module detects a bus reset.

After this module has detected a USB reset, a DVST interrupt occurs when the value of the RHST bits is fixed to 010 or 011.

32B.2.6 Test Mode Register

32B.2.6.1 USB Test Mode Register [TESTMODE] <Address: H'00C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UTST			
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	—	0	R/W	Reserved. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13, 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11	—	0	R/W	Reserved. When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
10 to 4	—	H'10	R	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	UTST	All 0	R/W	Test mode See the detailed description.

(1) Test mode (UTST) bits

When a value is written to these bits, this module outputs a USB test signal during high-speed operation.

Table 32B.8 lists the test modes of this module.

Table 32B.8 List of test mode operation

Test mode	Value of UTST Bits
Normal operation	0000
Test_J	0001
Test_K	0010
Test_SE0_NAK	0011
Test_Packet	0100
Test_Force_Enable	—
Reserved	0101 to 0111

Write a value to these bits according to the SetFeature request sent from the USB host during high-speed communication. When these bits contain a value from 0001 to 0100, this module does not enter the suspended state.

To perform a normal USB communication after setting a test mode, perform Power on reset.

32B.2.7 FIFO Port Registers

32B.2.7.1 CFIFO Port Register [CFIFO] <Address: H’014>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFOPORT (Low)															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

32B.2.7.2 CFIFO Port Register [CFIFO] <Address: H’016>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOPORT (High)															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFOPORT	All 0	R/W	FIFO port These bits are accessed to read received data from the FIFO buffer or to write send data to the FIFO buffer.

(1) FIFO port control bits (for FIFOPORT)

The send/receive buffer memory of this module has a FIFO structure (FIFO buffer). Use FIFO port registers to access the FIFO buffer. The FIFO port consists of the port register (CFIFO) to read data from and write data to the FIFO buffer, the register (CFIFOSEL) to select the pipe to be allocated to the FIFO port, and the control register (CFIFOCTR).

Individual FIFO ports have the following features:

1. The CFIFO port must be used to access the FIFO buffer through the DCP.
2. When functions specific to FIFO ports are used, the pipe number (selected pipe) specified in the CURPIPE bits cannot be changed.
3. The registers configured for a FIFO port do not affect any other FIFO ports.
4. The FIFO buffer memory can be accessed by either the CPU or SIE. Access by the CPU is not possible while the SIE has the right of access to the FIFO buffer memory.

(2) FIFO port bits (CFIFO)

When one of these registers is accessed, this module accesses the FIFO buffer allocated to the pipe number specified in the CURPIPE bits in the corresponding pipe select register (CFIFOSEL).

These registers can be accessed only when the FRDY bit of the respective control registers (CFIFOCTR) is 1 (or when the UCL_Dx_DREQ output is asserted by this module).

The valid bits of these registers vary depending on the values of the NBW and BIGEND bits. The valid bits are shown in **Table 32B.9** to **Table 32B.11**.

Table 32B.9 Endian Operation in 32-Bit Access (When MBW = 10)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	N + 3 address	N + 2 address	N + 1 address	N + 0 address
1	N + 0 address	N + 1 address	N + 2 address	N + 3 address

Table 32B.10 Endian Operation in 16-Bit Access (When MBW = 01)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*[1]		even-numbered address	odd-numbered address
1	even-numbered address	odd-numbered address	Writing: invalid Reading: prohibited*[1]	

Table 32B.11 Endian Operation in 8-Bit Access (When MBW = 00)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*[1]			Writing: valid Reading: valid
1	Writing: valid Reading: valid	Writing: invalid Reading: prohibited*[1]		

Note 1. Reading words or bytes from an invalid register is prohibited.

32B.2.7.3 CFIFO Port Select Register [CFIFOSEL] <Address: H'020>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	—	—	MBW		—	BIGEND	—	—	ISEL	—	CURPIPE			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R(0)/W	R	R	R/W	R/W	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read count mode This bit specifies DTLN read mode for the CFIFOCTR register. 0: Clears the DTLN bits when all received data is read. 1: Decrements the value of the DTLN bits each time received data is read.
14	REW	0	R(0)/W	Buffer pointer rewind Set this bit to 1 to rewind the buffer pointer. 0: Does not rewind the buffer pointer. 1: Rewinds the buffer pointer.
13, 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11, 10	MBW	All 0	R/W	CFIFO port access bit width This bit specifies the bit width for access to the CFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited
9	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
8	BIGEND	0	R/W	FIFO port byte endian control This bit specifies the byte endian of the CFIFO port. 0: Little endian 1: Big endian
7, 6	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
5	ISEL	0	R/W	FIFO port access direction with DCP selected This bit specifies the FIFO port access direction when DCP is selected for the CURPIPE bits. 0: Selects reading of buffer memory. 1: Selects writing of buffer memory.
4	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
3 to 0	CURPIPE	All 0	R/W	FIFO port access pipe specification This bit specifies the pipe number used for accessing the CFIFO port. 0000: DCP 0001: PIPE1 0010: PIPE2 ↓ 1000: PIPE8 1001: PIPE9 ↓ 1110: PIPE14 1111: PIPE15

(1) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the CFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the CFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(2) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).

Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.

If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(3) CFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the CFIFO port.

If you start reading after setting a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, do not change the value of these bits until data is all read.

Also, to set a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, temporarily change the original value of the CURPIPE bits to a different value, and then set the values of the CURPIPE and MBW bits at the same time.

For how to change the value of the CURPIPE bits, see the description of the CURPIPE bits.

When the pipe specified in the CURPIPE bits is in the sending direction, you cannot change the bit width from 8 bits to 16 bits or 32 bits or from 16 bits to 32 bits while writing to the buffer memory is in process.

Even with the 16-bit width or 32-bit width setting, you can write data also to odd bytes by using byte access control.

(4) FIFO port byte endian control (BIGEND) bit

This bit is used to specify the byte endian of the CFIFO port.

For details see **Section 32B.2.7.2(1), FIFO port control bits (for FIFOPORT)**.

(5) FIFO port access direction with DCP selected (ISEL) bit

To change the value of this bit when the specified pipe is DCP, write a value to this bit, read the bit, and then check that the written value is the same as the read value before proceeding to the next processing.

If the value of the bit is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

Set this bit simultaneously with setting of the CURPIPE bits.

(6) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number of the pipe through which to read or write data via the CFIFO port.

If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.

If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

32B.2.7.4 D0FIFO Port Select Register [D0FIFOSEL] <Address:H’028> D1FIFO Port Select Register [D1FIFOSEL] <Address:H’02C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW		—	—	—	—	—	—	CURPIPE			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R(0)/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read count mode This bit specifies Dx_FIFOCTR DTLN read mode. 0: Clears the DTLN bits when all received data is read. 1: Decrements the value of the DTLN bits each time received data is read.
14	REW	0	R(0)/W	Buffer pointer rewind Set this bit to 1 to rewind the buffer pointer. 0: Does not rewind the buffer pointer. 1: Rewinds the buffer pointer.
13	DCLRM	0	R/W	Automatic buffer memory clear mode after reading data through the specified pipe This bit prohibits or enables automatic buffer memory clear after data is read through the specified pipe. 0: Automatic FIFO buffer clear prohibited 1: Automatic FIFO buffer clear enabled
12	DREQE	0	R/W	UCL_Dx_DREQ output enable This bit prohibits or enables the output of the UCL_Dx_DREQ signal. 0: Output prohibited 1: Output enabled
11, 10	MBW	All 0	R/W	DxFIFO port access bit width This bit specifies the bit width for access to the DxFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited
9 to 4	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	CURPIPE	All 0	R/W	FIFO port access pipe specification 0000: No specification 0001: PIPE1 0010: PIPE2 ↓ 1000: PIPE8 1001: PIPE9

(1) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the DxFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the DxFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(2) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).

Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.

If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(3) Automatic FIFO buffer clear enable (DCLRM) bit

This bit is used to enable or disable the mode to automatically clear the FIFO buffer memory after reading data from the specified pipe. With this bit set, this module performs the “BCLR = 1” processing on the FIFO buffer, if a zero-length packet is received when the FIFO buffer allocated to the specified pipe is empty or if data reading ends because a short packet is received when the BFRE bit is 1.

Always clears this bit when you use this module with the BRDYM bit set.

(4) UCL_Dx_DREQ output enable (DREQE) bit

This bit is used to enable or disable the output of the UCL_Dx_DREQ signal.

When enabling the UCL_Dx_DREQ signal, set this bit always after setting a value in the CURPIPE bits. When changing the value of the CURPIPE bits, change the value always after clearing this bit.

(5) DxFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the DxFIFO port. For details, see **Section 32B.2.7.3(3), CFIFO port access bit width (MBW) bits**.

(6) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number through which to read or write data via the DxFIFO port.

If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.

Do not specify the same pipe number for the CURPIPE bits in the CFIFOSEL, D0FIFOSEL, and D1FIFOSEL registers. If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

32B.2.7.5 CFIFO Port Control Register [CFIFOCTR] <Address: H'022>
D0FIFO Port Control Register [D0FIFOCTR] <Address: H'02A>
D1FIFO Port Control Register [D1FIFOCTR] <Address: H'02E>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	—	DTLN											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W(1)	R(0)/W(1)	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	BVAL	0	R/W(1)	Buffer memory valid flag Specify 1 for this bit when writing to the FIFO buffer on the CPU side through the pipe specified in CURPIPE (current pipe) ends. 0: Invalid 1: Writing complete
14	BCLR	0	R(0)/W(1)	CPU buffer clear Specify 1 for this bit to clear the FIFO buffer on the CPU side of the current pipe. 0: Invalid 1: CPU buffer memory clear
13	FRDY	0	R	FIFO port ready This bit indicates whether the FIFO port can be accessed. 0: The FIFO port cannot be accessed. 1: The FIFO port can be accessed.
12	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
11 to 0	DTLN	All 0	R	Receive data length These bits indicate the length of receive data.

(1) Buffer memory valid flag (BVAL)

When the pipe (selected pipe) specified in the CURPIPE bits is in the sending direction, this bit must be set in the cases descried below. This module switches the FIFO buffer from the CPU side to the SIE side to enable data sending.

(1) To send short packets, set this bit when data writing ends.

(2) To send zero length packets, set this bit before writing data to the FIFO buffer.

(3) Set this bit after writing, to the pipe in continuous transfer mode, the data of which the size is a positive integral multiple of the maximum packet size and less than the buffer size.

When the maximum packet size of data is written to the pipe in non-continuous transfer mode, this module sets this bit to switch the FIFO buffer from the CPU side to the SIE side and enable data sending.

When this bit and the BCLR bit are set at the same time when the specified pipe is in the sending direction, this module clears the data that has been written so far and enables zero-length packets to be sent.

Setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit. Do not set this bit when the specified pipe is in the receiving direction.

(2) CPU buffer clear (BCLR) bit

When this bit is set, this module clears the FIFO buffer on the CPU side among the FIFO buffers allocated to the specified pipe.

Even if the two FIFO buffers in a double-buffer configuration are allocated to the specified pipe and the both buffers can be read, this module clears only one of the two buffers.

If this bit is set when the specified pipe is the DCP, this module clears the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or it is on the SIE side. To clear the buffer on the SIE side, set this bit always after setting the PID bits for DCP to “NAK”.

When the specified pipe is other than the DCP, setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit.

(3) FIFO port ready (FRDY) bit

This bit indicates whether the FIFO port can be accessed from the CPU. This bit is operated by this module.

In the cases described below, even when this module sets this bit, data cannot be read from the FIFO port because the FIFO buffer does not contain data to be read. In these cases, set the BCLR bit to clear the FIFO buffer to enable the next sending and receiving of data.

(1) A zero-length packet has been received while the FIFO buffer allocated to the specified pipe is empty.

(2) A short packet has been received and data reading has ended while the BFRE bit is 1.

(4) Receive data length (DTLN) bits

These bits indicate the length of receive data. These bits are operated by this module. During reading of the FIFO buffer, the value of these bits varies depending on the value of the RCNT bit as described below.

(1) When the RCNT bit is 0:
This module indicates a receive data length by using these bits until the CPU ends reading all received data from one FIFO buffer.
When the BFRE bit is 1, this module retains the receive data length until the BCLR bit is set even if reading of received data has ended.

(2) When the RCNT bit is 1:
This module decrements the value of these bits each time the CPU reads data.
(The value is decremented by 1 when the MBW bits are 00, by 2 when the MBW bits are 01 or by 4 when the MBW bits are 10.)

When the CPU ends reading from one FIFO buffer, this module clears these bits. If reading of one of the FIFO buffers in a double-buffer configuration, however, ends before reading of received data from the other FIFO buffer ends, these bits indicate the receive data length for the other FIFO buffer at the end of reading from the FIFO buffer of which reading ends earlier.

When these bits are read during reading of the FIFO buffer when the RCNT bit is 1, this module updates the value of these bits within 150 ns after a cycle of read access to the corresponding FIFO port.

32B.2.8 Interrupt Enable Registers (INTENBx, BRDYENB, NRDYENB, BEMPENB)

32B.2.8.1 Interrupt Enable Register 0 [INTENB0] <Address: H’030>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBSE	0	R/W	VBUS Interrupt Enable This bit prohibits or enables a USB interrupt when a VBINT interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
14	RSME	0	R/W	Frame number update interrupt enable Resume interrupt enable This bit prohibits or enables a USB interrupt when a RESM interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
13	SOFE	0	R/W	This bit prohibits or enables a USB interrupt when a SOF interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
12	DVSE	0	R/W	Device state transition interrupt enable This bit prohibits or enables a USB interrupt when a DVST interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
11	CTRE	0	R/W	Control transfer stage transition interrupt enable This bit prohibits or enables a USB interrupt when a CTRT interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
10	BEMPE	0	R/W	Buffer empty interrupt enable This bit prohibits or enables a USB interrupt when a BEMP interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
9	NRDYE	0	R/W	Buffer not ready response interrupt enable This bit prohibits or enables a USB interrupt when an NRDY interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
8	BRDYE	0	R/W	Buffer ready interrupt enable This bit prohibits or enables a USB interrupt when a BRDY interrupt is detected. 0: Interrupt output prohibited 1: Interrupt output enabled
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

32B.2.8.2 BRDY Interrupt Enable Register [BRDYENB] <Address: H'036>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBRDYE									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBRDYE	All 0	R/W	Pipe BRDY interrupt enable These bits prohibit or enable the BRDY bit to be set when a BRDY interrupt to a pipe is detected. 0: Interrupt output prohibited 1: Interrupt output enabled

(1) Pipe BRDY interrupt enable (PIPEBRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register, sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBRDY bits in the BRDYSTS register is 1 and software changes the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

32B.2.8.3 NRDY Interrupt Enable Register [NRDYENB] <Address: H’038>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPENRDYE									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPENRDYE	All 0	R/W	Pipe NRDY interrupt enable These bits prohibit or enable the NRDY bit to be set when a BRDY interrupt to a pipe is detected. 0: Interrupt output prohibited 1: Interrupt output enabled

(1) Pipe NRDY interrupt enable (PIPENRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPENRDY bit in the NRDYSTS register, sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPENRDY bits in the NRDYSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

32B.2.8.4 BEMP Interrupt Enable Register [BEMPENB] <Address: H’03A>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBEMPE									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBEMPE	All 0	R/W	Pipe BEMP interrupt enable These bits prohibit or enable the BEMP bit to be set when a BRDY interrupt to a pipe is detected 0: Interrupt output prohibited 1: Interrupt output enabled

Note: Bit numbers correspond to pipe numbers.

(1) Pipe BEMP interrupt enable (PIPEBEMPE) bits

When this module detects a BEMP interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBEMP bit in the BEMPSTS register, sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBEMP bits in the BEMPSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

32B.2.9 SOF Control Register

32B.2.9.1 SOF Pin Configuration Register [SOFCFG] <Address: H'03C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	BRDYM	—	—	SOFM		—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
6	BRDYM	0	R/W	PIPEBRDY interrupt status clear timing This bit specifies the timing at which the PIPEBRDY interrupt is to be cleared. 0: Software clears the status. 1: Hardware clears the status by reading from or writing to the FIFO buffer. This bit can be set only during initialization (before communication). The setting cannot be changed after communication.
5, 4	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
3, 2	SOFM	All 0	R/W	SOF function setting These bits are used to select SOF pulse output mode. 00: SOF output disabled 01: SOF output in units of 1 ms 10: μSOF output in units of 125 μs 11: Reserved
1, 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

32B.2.10 Interrupt Status

32B.2.10.1 Interrupt Status Register 0 [INTSTS0] <Address: H'040>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ			VALID	CTSQ		
Initial Value	0	0	0	0	0	0	0	0	—	0	0	0	0	0	0	0
	—	—	—	1	—	—	—	—	—	0	0	1	—	—	—	—
R/W	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R	R	R	R	R	R	R	R/W(0)	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBINT	0	R/W(0)	Change detection interrupt status This bit indicates the VBUS change detection interrupt status. 0: No VBUS interrupt is generated. 1: A VBUS interrupt is generated.
14	RESM	0	R/W(0)	Resume interrupt status This bit indicates the resume detection interrupt status. 0: No resume interrupt is generated. 1: A resume interrupt is generated.
13	SOFR	0	R/W(0)	Frame number update interrupt status This bit indicates the frame number update interrupt status. 0: No SOF interrupt is generated. 1: A SOF interrupt is generated.
12	DVST	0	R/W(0)	Device state transition interrupt status This bit indicates the device state transition interrupt. 0: No device state transition interrupt is generated. 1: A device state transition interrupt is generated.
11	CTRT	0	R/W(0)	Control transfer stage transition interrupt status This bit indicates the status of a control transfer stage transition interrupt. 0: No control transfer stage transition interrupt is generated. 1: A control transfer stage transition interrupt is generated.
10	BEMP	0	R	BEMP interrupt status This bit indicates the BEMP interrupt status. 0: No BEMP interrupt is generated. 1: A BEMP interrupt is generated.
9	NRDY	0	R	NRDY interrupt status This bit indicates the NRDY interrupt status. 0: No NRDY interrupt is generated. 1: An NRDY interrupt is generated.
8	BRDY	0	R	BRDY interrupt status This bit indicates the BRDY interrupt status. 0: No BRDY interrupt is generated. 1: A BRDY interrupt is generated.
7	VBSTS	0	R	VBUS input status This bit indicates the VBUS pin input status. 0: The VBUS pin is at the low level. 1: The VBUS pin is at the high level.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	DVSQ	All 0	R	Device state These bits indicate the device state. 000: Powered state 001: Default state 010: Address state 011: Configured state 1xx: Suspended state
3	VALID	0	R/ W(0)	USB request reception This bit indicates whether USB request reception is detected. 0: Not detected 1: A setup packet is received.
2 to 0	CTSQ	All 0	R	Control transfer stage These bits indicate the control transfer stage. 000: Idle or setup stage 001: Control reading data stage 010: Control reading status stage 011: Control writing data stage 100: Control writing status stage 101: Control writing (No Data) status stage 110: Control transfer sequence error 111: Reserved

Note: When you want to clear the status indicated by the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0 to only the bit to be cleared and 1 to other bits. Do not write 0 to any status bit that is currently 0.

Note: This module detects a status change indicated by the VBINT or RESM bit in this register even while the clock is stopped (the SUSPM bit is 0), and reports an interrupt corresponding to the status bit if the interrupt is enabled. Clearing the interrupt status must be done after the clock enabled.

(1) VBUS change interrupt status (VBINT) bit

This module sets this bit when it detects a change of the level of the VBUS pin input (from the high level to low level, or vice versa). This module indicates the level of the VBUS pin input by the VBSTS bit. When a VBINT interrupt occurs, read the VBSTS bit several times to check for consistency and remove chattering.

(2) Resume interrupt status (RESM) bit

This module sets this bit when it is in the suspended state (DVSQ bits are 1XX) and detects a falling edge of the signal at the DP pin.

(3) Frame number update interrupt status (SOFR) bit

This module sets this bit under the following conditions:

This module sets this bit when the frame number is updated. (The frame number update interrupt is monitored at intervals of 1 ms.)

This module detects an SOFR interrupt based on SOF interpolation even when an SOF packet from the USB host is damaged.

(4) Device state transition interrupt status (DVST) bit

When this module detects a change of the device state, this module updates the value of the DVSQ bits, and sets this bit. When a device state transition interrupt occurs, clear the interrupt status before this module detects the next device state transition.

(5) Control transfer stage transition interrupt status (CTRT) bit

When this module detects a transition of control transfer stage, this module updates the value of the CTSQ bits and sets this bit.

When a control transfer stage transition interrupt occurs, clear the interrupt status before this module detects the next transition of control transfer stage.

(6) Buffer empty interrupt status (BEMP) bit

This module sets this bit when at least one of the PIPEBEMP bits in the BEMPSTS register corresponding to the pipes for which the PIPEBEMPE bit in the BEMPENB register is set (that is, when this module detects a BEMP interrupt to at least one of the pipes for which is enabled BEMP interrupt notification).

For the conditions to assert the PIPEBEMP status signal, see the description of the BEMPSTS register.

This module clears this bit when writes 0 to all the PIPEBEMP bits corresponding to the pipes for which a BEMP interrupt has been enabled by setting the PIPEBEMPE bit.

Cannot clear this bit even by writing 0.

(7) Buffer not-ready interrupt status (NRDY) bit

This module sets this bit when at least one of the PIPENRDY bits in the BNRDYSTS register corresponding to the pipes for which the PIPENRDYE bit in the NRDYENB register is set (that is, when this module detects an NRDY interrupt to at least one of the pipes for which is enabled NRDY interrupt notification).

For the conditions to assert the PIPENRDY status signal, see the description of the NRDYSTS register.

This module clears this bit when writes 0 to all the PIPENRDY bits corresponding to the pipes for which a NRDY interrupt has been enabled by setting the PIPENRDYE bit.

Software cannot clear this bit even by writing 0.

(8) Buffer ready interrupt status (BRDY) bit

This module sets this bit when at least one of the PIPEBRDY bits in the BRDYSTS register corresponding to the pipes for which the PIPEBRDYE bit in the BRDYENB register is set (that is, when this module detects an BRDY interrupt to at least one of the pipes for which is enabled BRDY interrupt notification).

For the conditions to assert the PIPEBRDY status signal, see the description of the BRDYSTS register.

This module clears this bit when writes 0 to all the PIPEBRDY bits corresponding to the pipes for which a BRDY interrupt has been enabled by setting the PIPEBRDYE bit.

Software cannot clear this bit even by writing 0.

32B.2.10.2 BRDY Interrupt Status Register [BRDYSTS] <Address: H'046>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBRDY									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBRDY	All 0	R/W(0)	Pipe BRDY interrupt status These bits indicate the BRDY interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register when the BRDYM bit is 0, write 0 to only the bit to be cleared and 1 to other bit.

Note: When the BRDYM bit is 0, clearing the BRDY interrupt status must be done always before the next access to the FIFO.

(1) Pipe BRDY interrupt status (PIPEBRDY) bit

When this module detects a BRDY interrupt to a pipe, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register. At that time, if already set the corresponding bit in the BRDYENB register, this module sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.

Conditions to generate and clear a BRDY interrupt vary depending on the values of the BRDYM bit, and the BFRE bit for each pipe.

(a) BRDYM = 0 and BFRE = 0

When the BRDYM and BFRE bits are 0, the BRDY interrupt is generated to indicate that the FIFO port has become ready for access.

Under the conditions described below, this module generates an internal BRDY interrupt request trigger, and sets the PIPEBRDY bit corresponding to the pipe for which the request trigger is generated.

1. For the pipe in the sending direction

(a) When the DIR bit changes from 0 to 1

(b) When this module has ended sending packets through a pipe when data writing by the CPU to the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
In continuous transfer mode, a request trigger is generated when data has all been sent from one FIFO buffer.

(c) When, in a double-buffer configuration, one FIFO buffer is empty when writing to the other FIFO buffer has ended
If sending to one FIFO buffer has ended during writing to the other FIFO buffer, no request trigger is generated until the ongoing writing to the other FIFO buffer ends.

(d) When this module flushes the FIFO buffer allocated to the pipe of which the transfer type is isochronous.

(e) When the state of the FIFO buffer is changed from the write-disabled state to the write-enabled state by writing 1 to the ACLRM bit

No request trigger is generated when the pipe is the DCP (in other words, when data is sent by a control transfer).

2. For the pipe in the receiving direction

(a) When this module has ended receiving packets through a pipe and reading from the FIFO buffer is enabled when data reading by the CPU from the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
No request trigger is generated for the transaction that involves a data PID mismatch.
In continuous transmission/reception mode, no request trigger is generated when the data size is the maximum packet size and the FIFO buffer still has a free space.
If a short packet is received, a request trigger is generated even when the FIFO buffer has a free space.
When the transaction counter is used, a request trigger is generated when the specified number of packets have been received.
In that case, the request trigger is generated even if the FIFO buffer has a free space.

(b) When, in a double-buffer configuration, one FIFO buffer is in the read-enabled state when reading from the other FIFO buffer has ended
If receiving from one FIFO buffer has ended during reading from the other FIFO buffer, no request trigger is generated until the ongoing reading from the other FIFO buffer ends.

This interrupt does not occur during the communication at the status stage of a control transfer.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to "0" by writing "0" to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to all the PIPEBRDY bits corresponding to other pipes.

Clearing the pipe BRDT interrupt status must be done always before the next access to the FIFO buffer.

(b) When BRDYM = 0 and BFRE = 1

When the BRDYM bit is 0 and the BFRE bit is 1, this module determines that a BRDY interrupt occurs when all the data for a transfer has been read through a receiving pipe, and sets the PIPEBRDY bit corresponding to the pipe.

This module determines that the last data in a transfer has been received when one of the following conditions is met:

1. A short packet or a zero-length packet has been received.
2. The transaction counter (TRNCNT bits) is used, and as many packets as the value of the TRNCNT bits have been received.

When one of the above conditions is met and reading of the relevant data has ended, this module determines that all the data in a transfer has been read.

If a zero-length packet is received when the FIFO buffer is empty, this module determines that all the data in a transfer has been read when the FRDY bit is set and the DTLN bits are cleared in the corresponding FIFO Port Control Register. To start the next transfer in that case, write 1 to the BCLR bit in the corresponding FIFOCTR.

When the BRDYM bit is 0 and the BFRE bit is 1, this module does not detect any BRDY interrupt to the pipe in the sending direction.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to “0” by writing “0” to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to the PIPEBRDY bits corresponding to other pipes.

In this mode, do not change the value of the BFRE bit until all the processing for a transfer ends.

If you need to change the value of the BFRE bit during the transfer, set the ACLRM bit to clear all the FIFO buffers for the specified pipe.

(c) When BRDYM = 1 and BFRE = 0

When the BRDYM bit is 1 and the BFRE bit is 0, the values of individual PIPEBRDY bits interlock with the values of the BSTS bits for individual pipes. In other words, this module sets or clears the BRDY interrupt status of a pipe according to the state of the FIFO buffer allocated to the pipe.

(a) For the pipe in the sending direction
This module sets the PIPEBRDY bit for the pipe when data can be written to the FIFO port or clears the PIPEBRDY bit when data cannot be written to the FIFO port.
The BRDY interrupt signal, however, is not asserted even when the sending pipe is write-enabled if the pipe is the DCP.

(b) For the pipe in the receiving direction
This module sets the PIPEBRDY bit for the pipe when data can be read from the FIFO port or clears the PIPEBRDY bit when all data has been read (that is, when data reading from the FIFO port is disabled).
If a zero-length packet is received when the FIFO buffer is empty, the PIPEBRDY bit corresponding to the specified pipe is kept being set and the BRDY interrupt signal is kept being asserted until sets the BCLR bit.

When the BRDYM bit is 1 and the BFRE bit is 0, this module cannot clear any PIPEBRDY bit.

When the BRDYM bit is 1, all the BFRE bits (for all pipes) must be 0.

32B.2.10.3 NRDY Interrupt Status Register [NRDYSTS] <Address: H'048>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPENRDY									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPENRDY	All 0	R/W(0)	Pipe NRDY interrupt status These bits indicate the NRDY interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register, write 0 only to the bit and 1 to all other bit.

(1) Pipe NRDY interrupt status (PIPENRDY) bit

When this module issues an internal NRDY interrupt request for a pipe of which the PID is set to BUF, this module sets the PIPENRDY bit corresponding to the pipe in the NRDYSTS register. At that time, if the NRDYENB register bit corresponding to the pipe is set, this module sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.

This module issues an internal NRDY interrupt request for individual pipes under the conditions described below. This module does not issue any interrupt request at the status stage of a control transfer.

(a) For the pipe in the sending direction

(1) When an IN token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer does not contain send data
When an IN token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.
If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module sends a zero- length packet, and sets the OVRN bit.

(b) For the pipe in the receiving direction

(1) When an OUT token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module issues an NRDY interrupt request, sets the PIPENRDY bit, and sets the OVRN bit.
If the transfer type of the pipe for which the interrupt is generated is not isochronous transfer, this module issues an NRDY interrupt request when sending an NAK Handshake signal after receiving the data that follows the OUT token, and sets the PIPENRDY bit.
Note, however, that this module does not issue an NRDY interrupt request when resending data (when a DATA-PID mismatch has occurred).
This module does not issue an NRDY interrupt request also when an error has occurred in a data packet.

(2) When a PING token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
When a PING token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.

(3) When the transfer type of the specified pipe is isochronous transfer, the PID bits corresponding to the specified pipe are 01 (BUF), and data has not been received normally within an interval frame
When an SOF token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit for the specified pipe.

32B.2.10.4 BEMP Interrupt Status Register [BEMPSTS] <Address: H’04A>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBEMP									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R/W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
9 to 0	PIPEBEMP	All 0	R/W(0)	Pipe BEMP interrupt status These bits indicate the BEMP interrupt status of each pipe. 0: No interrupt is generated. 1: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register, write 0 only to the bit and 1 to all other bit.

(1) Pipe BEMP interrupt status (PIPEBEMP) bit

When this module detects a BEMP interrupt to a pipe of which the PID is set to BUF, this module sets the PIPEBEMP bit corresponding to the pipe in the BEMPSTS register. At that time, if the BEMPENB register bit corresponding to the pipe is set, this module sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

This module issues an internal BEMP interrupt request for individual pipes under the conditions described below.

1. When data sending (including sending of zero-length packets) to a pipe in the sending direction has ended and the FIFO buffer allocated to the pipe is empty
   In a single-buffer configuration, this module generates a BRDY interrupt at the same time as issuing an internal BEMP interrupt request for the pipes other than the DCP.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:

(a) When software (DMAC) has already started writing to the CPU-side FIFO buffer, in a double-buffer configuration, when sending data for one buffer ends

(b) When the buffer is cleared (to empty the buffer) by writing 1 to the ACLRM or BCLR bit

(c) During an IN transfer (sending zero-length packets) at the status stage of a control transfer

2. For the pipe in the receiving direction
   When the data larger than the specified maximum packet size has been received normally
   In that case, this module issues a BEMP interrupt request, sets the PIPEBEMP bit for the specified pipe, discards the received data, and changes the value of the PID bits for the specified pipe to 11 (STALL).
   Then, this module returns a STALL packet.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:

(a) When a CRC or bit stuff error has been detected in the received data

(b) When a SETUP transaction is being executed

Writing 0 to this bit clears the interrupt status.

Writing 1 to this bit causes no effect.

32B.2.11 Frame Number Registers (FRMNUM, UFRMNUM)

32B.2.11.1 Frame Number Register [FRMNUM] <Address: H’04C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	—	—	—	FRNM										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W(0)	R/W(0)	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	OVRN	0	R/W(0)	Overrun/underrun detection status This bit indicates whether an overrun or underrun is detected in the pipe being used to perform an isochronous transfer. 0: No error 1: An error occurred.
14	CRCE	0	R/W(0)	CRC error detection status This bit indicates the CRC error detection status for the pipe being used to perform an isochronous transfer. 0: No error 1: An error occurred.
13 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10 to 0	FRNM	All 0	R	Frame number These bits indicate the latest frame number.

Note: The OVRN bit is intended for debugging. When designing a system, design transfer timings appropriately to prevent buffer overruns and underruns.

(1) Overrun/underrun detection status (OVRN) bit

This module sets this bit when this module detects an overrun or underrun in a pipe of which the transfer type is isochronous transfer.

When this module detects an overrun or underrun, this module issues an internal NRDY interrupt request. For details, see **Section 32B.2.10.3(1), Pipe NRDY interrupt status (PIPENRDY) bit**.

Software can clear this bit by writing 0 to this bit. If the CRCE bit should not be cleared together when this bit is cleared, write H’40.

When the peripheral controller function is selected

This module sets this bit in the following cases:

1. When an IN token is received although writing send data to the FIFO has not ended, in the case of a pipe that is in the sending direction and performs an isochronous transfer
2. When an OUT token is received although the free space of FIFO buffer is less than the size of one FIFO buffer, in the case of a pipe that is in the sending direction and performs an isochronous transfer

(2) CRC error detection status (CRCE) bit

This module sets this bit when this module detects a CRC or bit stuff error in a pipe of which the transfer type is isochronous transfer.

Software can clear this bit by writing 0 to this bit. If the OVRN bit should not be cleared together when this bit is cleared, write H’80.

When this module detects a CRC error, this module issues an internal NRDY interrupt request. For details, see **Section 32B.2.10.3(1), Pipe NRDY interrupt status (PIPENRDY) bit**.

(3) Frame number (FRNM) bits

This module updates the value of these bits each time an SOF packet is received (once per 1 ms), and indicates the latest frame number in these bits.

When reading these bits, read them twice and check for consistency.

32B.2.11.2 Micro Frame Number Register [UFRMNUM] <Address: H’04E>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to these bits. Fix these bits to 0.
14 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2 to 0	UFRNM	All 0	R	Micro frame These bits indicate the micro frame number.

(1) Micro frame number (UFRNM) bit

In high-speed transfer mode, this module writes the micro frame number to these bits. In a mode other than high-speed transfer mode, this module writes H’00 to these bits.

When reading these bits, read them twice and check for consistency.

32B.2.12 USB Address

32B.2.12.1 USB Address Register [USBADDR] <Address: H’050>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	USBADDR						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10 to 8	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
7	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
6 to 0	USBADDR	All 0	R	USB address These bits indicate the USB address allocated by the host.

(1) USB address (USBADDR) bits

When this module has received and normally processed a SetAddress request, this module writes the received USB address to these bits.

When this module detects a USB bus reset, this module writes H’00 to these bits.

32B.2.13 USB Request Registers

USB request registers are used to store control transfer setup requests.

These registers store the values set in the received USB requests.

32B.2.13.1 USB Request Type Register [USBREQ] <Address: H’054>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	bRequest								bmRequestType							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	bRequest	All 0	R	Request Value of USBRequestbRequest
7 to 0	bmRequestType	All 0	R	Request type Value of USBRequestbmRequestType

(1) USB request (bRequest) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

(2) USB request type (bRmRequestType) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.13.2 USB Request Value Register [USBVAL] <Address: H’056>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wValue															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wValue	All 0	R	Value Value of the wValue field in a USB request

(1) Value (wValue) bits

These bits indicate the value of the wValue field in a USB request.

These bits indicate the value of wValue field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.13.3 USB Request Index Register [USBINDX] <Address: H’058>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wIndex															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wIndex	All 0	R	Index Value of the wIndex field in a USB request

(1) Index (wIndex) bits

These bits indicate the value of the wIndex field in a USB request.

These bits indicate the value of wIndex field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.13.4 USB Request Length Register [USBLENG] <Address: H’05A>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wLength															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wLength	All 0	R	Length Value of the wLength field in a USB request

(1) Length (wLength) bits

These bits indicate the value of the wLength field in a USB request.

These bits indicate the value of wLength field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

32B.2.14 DCP Configuration

When performing a data communication by a control transfer, use the default control pipe (DCP).

32B.2.14.1 DCP Configuration Register [DCPCFG] <Address: H’05C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CNTMD	SHTNAK	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	—	—	—	—	—	—	—	R/W	R/W	—	—	—	—	—	—	—

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0		Nothing is assigned to these bits. Fix these bits to 0.
8	CNTMD	0	R/W	Continuous transfer mode This bit specifies whether to use the default control pipe to perform communication in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode
7	SHTNAK	0	R/W	Pipe disable at transfer end When the default control pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0: Continues the pipe when the transfer ends 1: Disables the pipe when the transfer ends
6 to 0	—	All 0		Nothing is assigned to these bits. Fix these bits to 0.

32B.2.14.2 DCP Max. Packet Size Register [DCPMAXP] <Address: H’05E>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	MXPS						
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
6 to 0	MXPS	H’40	R/W	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the DCP.

(1) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the DCP.

The default is H’40 (64 bytes).

The value of the MXPS bits must conform to the USB Specification.

Writing a value to the MXPS bits must be done when the PID is set to NAK and no value is set in the CURPIPE bits. If you need to change the value of these bits after changing the PID setting for the specified pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of these bits. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

32B.2.14.3 DCP Control Register [DCPCTR] <Address: H'060>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	—	—	—	—	—	SQCLR	SQSET	SQMON	PBUSY	—	—	CCPL	PID	
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W	R	R	R	R	R	R	R	R(0)/W(1)	R(0)/W(1)	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer status This bit indicates the accessibility status of the DCP FIFO buffer. 0: Buffer access is not possible. 1: Buffer access is possible.
14 to 9	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
8	SQCLR	0	R(0)/ W(1)	Toggle bit clear This bit can set DATA0 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: Writing disabled 1: DATA0 specified
7	SQSET	0	R(0)/ W(1)	Toggle bit set This bit can set DATA1 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: Writing disabled 1: DATA1 specified
6	SQMON	0	R	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0: DATA0 1: DATA1
5	PBUSY	0	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.
4 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2	CCPL	0	R/W	Control transfer end enable Setting this bit permits the status stage of the control transfer to end. 0: Does not permit the control transfer to end. 1: Permits the control transfer to end.
1, 0	PID	All 0	R/W	Response PID These bits control responses from this module in control transfer. 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response

(1) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the DCP. This bit is operated by this module. The meaning of this bit varies as follows depending on the value of the ISEL bit:

1. When ISEL = 0: This bit indicates whether receive data can be read from the FIFO buffer.
2. When ISEL = 1: This bit indicates whether send data can be written to the FIFO buffer.

(2) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.

Do not set the SQCLR and SQSET bits at the same time.

Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.

If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(3) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.

Do not set the SQCLR and SQSET bits at the same time.

Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.

If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(4) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the specified pipe. This bit is operated by this module. When a transaction ends normally, this module toggles this bit.

This module, however, does not toggle this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.

When a SETUP packet is received normally, this module sets this bit (to specify DATA1 as the expected value).

This module does not reference this bit in an IN or OUT transaction at the status stage. Also, this module does not toggle this bit even when the transaction ends normally.

(5) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the specified pipe starts. This module changes this bit from 1 to 0 when the transaction ends.

Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(6) Control transfer end enable (CCPL) bit

When software sets this bit when the PID of the specified pipe is BUF, this module ends the status stage of the ongoing control transfer.

In other words, in a control read transfer, this module sends an ACK handshake in response to an OUT transaction request from the USB host, and, in a control write or no-data control transfer, this module sends a zero-length packet in response to an IN transaction request from the USB host. If, however, a SetAddress request is detected, this module performs automatic response throughout the period from the setup stage to the end of the status stage regardless of the value of this bit.

When a new SETUP packet is received, this module changes this bit from 1 to 0.

When the VALID bit is 1, software cannot set this bit.

(7) Response PID (PID) bits

The setting of these bits must be changed from NAK to BUF when the data stage or status stage of a control transfer is executed.

This module changes the value of these bits in the following cases:

1. This module changes the value of these bits to 00 (NAK) when it receives a SETUP packet. At that time this module sets the VALID bit. Software cannot change the value of these bits until it clears the VALID bit.
2. When set these bits to 01 (BUF), this module changes the value of these bits to 11 (STALL) when it receives the data exceeding the specified maximum packet size.
3. This module changes the value of these bits to 1x (STALL) when it detects a sequence error in a control transfer.
4. This module changes the value of these bits to 00 (NAK) when it detects a USB reset.

This module does not reference these bits during SetAddress request processing (automatic processing).

32B.2.15 Pipe Configuration Registers (PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI)

To configure pipes 1 to 15, use the PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPExCTR, PIPExTRE, and PIPExTRN registers.

Select the pipes to be used by using the PIPESEL register, and then configure functions of individual pipes by using the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers. Note that you can use the PIPExCTR, PIPExTRE, and PIPExTRN registers for setting regardless of the pipe selection by the PIPESEL register.

32B.2.15.1 Pipe Window Select Register [PIPESEL] <Address: H’064>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	PIPESEL	All 0	R/W	Pipe window select These bits specify a pipe for registers at addresses H’68 to H’6E. 0000: No selection 0001: PIPE1 0010: PIPE2 0011: PIPE3 0100: PIPE4 0101: PIPE5 0110: PIPE6 0111: PIPE7 1000: PIPE8 1001: PIPE9

Note: When the PIPESEL bits are 0000, all bits of the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers are cleared.
When the PIPESEL bits are 0000, writing to the registers at addresses H’68 to H’6E is ignored.

(1) Pipe Window select (PIPESEL) bits

When software a value from 0001 to 1111 to these bits, this module indicate the pipe information and settings corresponding to the registers at addresses H’68 to H’6E. After a pipe is selected by these bits, the values set in the areas at addresses H’68 to H’6E are applied, by this module, to the transfer operation using the selected pipe.

When writes 0000 to these bits, this module writes 0 to all bits of the registers at addresses H’68 to H’6E. Then, writing to the areas at addresses H’68 to H’6E is ignored.

32B.2.15.2 Pipe Configuration Register [PIPECFG] <Address: H'068>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE		—	—	—	BFRE	DBLB	CNTMD	SHTNAK	—	—	DIR	EPNUM			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	TYPE	All 0	R/W	Transfer type These bits specify the transfer type of the pipe specified in the PIPESEL bit. 00: The pipe cannot be used. 01: Bulk transfer 10: Interrupt transfer 11: Isochronous transfer
13 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10	BFRE	0	R/W	BRDY interrupt operation specification This bit specifies the timing at which this module notifies a BRDY interrupt relating to the specified pipe. 0: A BRDY interrupt is notified when data is sent or received. 1: A BRDY interrupt is notified when reading of data is completed.
9	DBLB	0	R/W	Double-buffer mode This bit specifies a single or double FIFO buffer to be used by the specified pipe. 0: Single buffer 1: Double buffer
8	CNTMD	0	R/W	Continuous transfer mode This bit specifies whether to use the specified pipe to perform communication in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode
7	SHTNAK	0	R/W	Pipe disable at transfer end When the specified pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0: Continues the pipe when the transfer ends. 1: Disables the pipe when the transfer ends.
6, 5	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
4	DIR	0	R/W	Transfer direction This bit specifies the transfer direction of the specified pipe. 0: Receiving direction 1: Sending direction
3 to 0	EPNUM	All 0	R/W	Endpoint number These bits specify the endpoint number of the specified pipe.

(1) Transfer type (TYPE) bits

These bits are used to specify the USB transfer type of the pipe (selected pipe) specified in the PIPESEL bits.

Table 32B.12 lists pipes and the transfer types specifiable in these bits.

Table 32B.12 Selected Pipes and the Transfer Types Specifiable in the TYPE Bits

Selected Pipe	TYPE Bits	USB Transfer Type
PIPE1 or PIPE2	01 or 11	Bulk or isochronous transfer
PIPE3 to PIPE5	01	Bulk transfer
PIPE6 to PIPE9	10	Interrupt transfer

Always specify a value other than 00 in these bits for a selected pipe before setting the PID of the selected pipe to BUF (to start USB communication using the selected pipe).

The value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK. If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(2) BRDY interrupt operation specification (BFRE) bit

This bit is valid when the selected PIPE is PIPE1 to PIPE5.

When set this bit and been using the selected pipe in the receiving direction (in other words, the DIR bit is 0), this module detects the end of transfer (when it occurs) and generates a BRDY interrupt when reading of the last packet ends.

If a BRDY interrupt occurs with the above settings, software must write 1 to the BCLR bit. The FIFO buffer allocated to the selected pipe remains unready for reception until 1 is written to the BCLR bit.

When set this bit and been using the selected pipe in the sending direction (in other words, the DIR bit is 1), this module does not generate any BRDY interrupt.

For details, see the description of the PIPEBRDY interrupt status bit.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(3) Double-buffer mode (DBLB) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5.

When set this bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in the BUFSIZE bits in the PIPEBUF register.

The size of the FIFO buffer this module allocates to the selected pipe is as follows:

$$(\text{BUFSIZE} + 1) \times 64 \times (\text{DBLB} + 1)\ [\text{bytes}]$$

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(4) Continuous transfer mode (CNTMD) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5, and the transfer type of the selected pipe is bulk transfer.

This module determines whether data sending from or receiving in the FIFO buffer allocated to the selected pipe has ended according to the value of this bit in the way described in **Table 32B.13**.

Table 32B.13 How to Determine the End of Data Sending from or Receiving in the FIFO Buffer According to the Value of the CNTMD Bit

CNTMD Bit Setting Value	How to Determine Whether Reading or Sending is Enabled
0	Condition for enabling reading from the FIFO buffer when the receiving direction is set ("DIR = 0"): This module receives one packet
	Condition for enabling sending from the FIFO buffer when the sending direction is set ("DIR = 1"): Either of the following conditions (1) and (2) is met: (1) Data for the maximum packet size is written to the FIFO buffer. (2) Data for the short packet (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit.
1	Conditions for enabling reading from the FIFO buffer when the receiving direction is set ("DIR = 0"): (1) The number of bytes in data received in the FIFO buffer allocated to the selected pipe becomes equal to the number of allocated bytes ((BUFSIZE + 1) * 64). (2) This module receives a short packet other than a zero-length packet. (3) This controller receives a zero-length packet when the FIFO buffer allocated to the selected pipe already contains data. (4) Packets are received as many times as the value of the transaction counter set for the selected pipe.
	Condition for enabling sending from the FIFO buffer when the sending direction is set ("DIR = 1"): One of the following conditions (1) to (3) is met: (1) The amount of written data becomes equal to the size of one FIFO buffer allocated to the selected pipe. (2) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit. (3) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer, and a transfer end signal is asserted at the same time of the last writing.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(5) PIPE disable at transfer end (SHTNAK) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5 and is in the receiving direction.

When set this bit for a selected pipe in the receiving direction, this module changes the PID of the selected pipe to NAK when this module determines the end of data transfer to the selected pipe. This module determines the end of transfer when one of the following conditions (1) and (2) is met:

(1) This module has normally received short packet data (including zero-length packets).

(2) When using a transaction counter, this module has normally received as many packets as the value set in the transaction counter.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

For the pipes in the sending direction, this bit must be cleared.

(6) Transfer direction (DIR) bit

When writes 0 to this bit for a selected pipe, this module uses the selected pipe in the receiving direction. When software writes 1 to this bit for the selected pipe, this module uses the selected pipe in the sending direction.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(7) Endpoint number (EPNUM) bits

These bits are used to specify the endpoint number of the endpoint of a selected pipe. Note that specifying 0000 in these bits for a pipe means that the pipe is not used.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

The combination of the values of the DIR bit and EPNUM bits for a pipe must be unique among those for all pipes.
(The setting "EPNUM = 000" [the selected pipe is not used] can be duplicated for multiple pipes.)

32B.2.15.3 Pipe Buffer Setting Register [PIPEBUF] <Address: H’06A>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BUFSIZE					—	—	BUFNMB							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14 to 10	BUFSIZE	All 0	R/W	Buffer size These bits specify the size of the FIFO buffer for the pipe specified in the PIPESEL bit. H’00: 64 bytes H’01: 128 bytes ... (H’1F: 2 Kbytes)
9, 8	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
7 to 0	BUFNMB	All 0	R/W	Buffer number These bits specify the FIFO buffer number of the specified pipe. (H’4 to H’7F)

Note: Changing values of these register bits for a selected pipe must be done when sets the PID of the selected pipe to NAK, and specifies no pipe in the CURPIPE bits.

Note: If you change values of these register bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the values of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(1) Buffer size (BUFSIZE) bits

These bits are used to specify the size of the FIFO buffer to be allocated to the selected pipe. Specify the FIFO buffer size in units of blocks. One block has 64 bytes.

When set the DBLB bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in these bits.

The size of the FIFO buffer this module allocates to the selected pipe is as follows:

$$(\text{BUFSIZE} + 1) \times 64 \times (\text{DBLB} + 1) \text{ [bytes]}$$

The following value can be specified in these bits:

(1) Any value from H’0 to H’1F when the selected pipe is PIPE1 to PIPE5.

(2) H’0 only when the selected pipe is PIPE6 to PIPE9.

In continuous transfer mode (CNTMD = 1), specify an integral multiple of the maximum packet size in the BUFSIZE bits.

(2) Buffer number (BUFNMB) bits

These bits are used to specify the block number of the first block in the FIFO buffer to be allocated to the selected pipe. This module allocates the following blocks of FIFO buffer to the selected pipe:

Block with block number “BUFNMB” to the block with block number “BUFNMB + (BUFSIZE + 1) × (DBLB + 1) - 1”

The value of these bits must be H’04 to H’7F. Note, however, that the following rules must be observed: Value " H’00" is exclusively used for DCP.

Value “H’04” is exclusively used for PIPE6. When, however, PIPE6 is not used, this value can be used for another pipe. If PIPE6 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’04 in the BUFNMB bits for PIPE6.

Value “H’05” is exclusively used for PIPE7. When, however, PIPE7 is not used, this value can be used for another pipe. If PIPE7 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’05 in the BUFNMB bits for PIPE7.

Value “H’06” is exclusively used for PIPE8. When, however, PIPE8 is not used, this value can be used for another pipe. If PIPE8 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’06 in the BUFNMB bits for PIPE8.

Value “H’07” is exclusively used for PIPE9. When, however, PIPE9 is not used, this value can be used for another pipe. If PIPE9 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets H’07 in the BUFNMB bits for PIPE9.

32B.2.15.4 Pipe Maximum Packet Size Register [PIPEMAXP] <Address: H’06C>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	MXPS										
Initial Value	0	0	0	0	0	0	0	0	0	0 (1)	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10 to 0	MXPS	*Note	R/W	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the specified pipe. For PIPE6 to PIPE8, a value from H’1 to H’40 (bytes) can be set.

Note: The initial value of the MXPS bits is H’00 when no pipe is specified in the PIPESEL bits in the PIPESEL register or H’40 when a pipe is specified in the PIPESEL bits.

(1) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the selected pipe.

The initial value of these bits is H’40 (64 bytes).

- For PIPE1 and PIPE2, a value from H’1 (1 byte) to H’400 (1024 bytes) can be specified.
- For PIPE3 to PIPE5, H’8 (8 bytes), H’10 (16 bytes), H’20 (32 bytes), H’40 (64 bytes), or H’200 (512 bytes) can be specified. (Bits [2:0] are excluded.)
- For PIPE6 to PIPE9, a value from H’1 (1 byte) to H’40 (64 bytes) can be specified.

The value of the MXPS bits for individual transfer type must conform to the USB Specification.

Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

32B.2.15.5 Pipe Cycle Control Register [PIPEPERI] <Address: H’06E>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
12	IFIS	0	R/W	Isochronous IN buffer flush This bit specifies whether to perform a buffer flush when the pipe specified in the PIPESEL bit is used for isochronous IN transfer. 0: Does not perform a buffer flush. 1: Performs a buffer flush.
11 to 3	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
2 to 0	IITV	All 0	R/W	These bits specify the transfer interval of the specified pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.

(1) Isochronous IN buffer flush (IFIS) bit

When the selected pipe is used for isochronous IN transfer, this bit is used to specify this module automatically clears the FIFO buffer if this module fails to receive the IN token from the USB host in a (micro) frame sent at intervals specified in the IITV bits.

In double-buffer mode (DBLB = 1), this module clears only the data in one buffer used earlier than the other.

This module clears the FIFO buffer when it receives an SOF packet immediately after the (micro) frame in which the IN token hag to be received. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

(2) Interval error detection interval (IITV) bits

These bits specify the interval of interval error detection for the selected pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.

Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

If you change the value specified in these bits to another after a USB communication, change the PID to NAK, and then set the ACLRM bit to initialize the interval timer before changing the value.

For PIPE3 to PIPE5 these bits are ignored. Write 0 to all these bits corresponding to PIPE3 to PIPE5.

You can specify a value in these bits when the transfer type of the selected pipe is isochronous.

(a) When the selected pipe is used for isochronous OUT transfer

If this module does not receive any data packet in the (micro) frame sent at intervals specified in the IITV bits, this module generates an NRDT interrupt.

This module generates an NRDY interrupt also if this module cannot receive data because an error, e.g., CRC error, has occurred in a data packet or because the FIFO buffer is full (such a situation might result if, for example, software [DMAC] delays in reading data from the FIFO buffer).

This module generates the NRDY interrupt when it receives an SOF packet. Even if the SOF packet is corrupted, this module generates the NRDY interrupt in the same timing to receive the SOF packet by the use of the internal interpolation function.

When, however, the value of the IITV bits is not 0, this module generates the NRDY interrupt every time an SOF packet is received at the specified intervals after interval counting starts.

If the PID of the selected pipe is changed to NAK after the interval timer starts, this module does not generate the NRDY interrupt even when it receives an SOF packet.

The condition for starting interval counting varies by the value of the IITV bits.

(a) When IITV = 0: Interval counting starts when the PID of the selected pipe is changed to BUF.

(Micro) frame	SOF	SOF	SOF	OUT	DATA0	SOF	OUT	DATA0
Setting of PID bits	NAK	NAK	BUF			BUF		
Whether token reception is expected (0: reception expected —: non-reception expected)	—	—	0			0		
Start of interval counting			↑					

Figure 32B.1 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 0

(b) When IITV is not 0: Interval counting starts at the end of the first normal reception of data packet after the PID of the selected pipe is changed to BUF.

(Micro) frame	SOF	SOF	SOF OUT DATA0	SOF	SOF OUT DATA0	SOF	SOF OUT DATA0
Setting of PID bits	NAK	BUF	BUF	BUF	BUF	BUF	BUF
Whether token reception is expected (0: reception expected —: non-reception expected)	—	—	0	—	0	—	0
Start of interval counting			↑				

Figure 32B.2 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 1

(b) When the selected pipe is used for isochronous IN transfer

The IITV bits are used in combination with the setting of the IFIS bit to 1. When the IFIS bit is 0, this module sends a data packet in response to a received token regardless of the value of the IITV bits.

When the IFIS bit is 1, if this module does not receive any IN token in the (micro) frame sent at intervals specified in the IITV bits although the FIFO buffer has sendable data, this module clears the FIFO buffer.

This module clears the FIFO buffer also when it cannot receive an IN token normally because of a bus error, e.g., CRC error.

This module clears the FIFO buffer when it receives an SOF packet. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

The condition for starting interval counting varies by the value of the IITV bits. (The condition is the same as that for isochronous OUT transfer.)

The interval counter is cleared when one of the following conditions (1) to (3) is met:

(1) This module is reset (then, also the IITV bits are cleared).

(2) The ACLRM bit is set.

(3) This module detects a USB bus reset.

32B.2.16 Pipe Control Registers (PIPExCTR)

32B.2.16.1 PIPE1 Control Register [PIPE1CTR] <Address: H’070> PIPE2 Control Register [PIPE2CTR] <Address: H’072> PIPE3 Control Register [PIPE3CTR] <Address: H’074> PIPE4 Control Register [PIPE4CTR] <Address: H’076> PIPE5 Control Register [PIPE5CTR] <Address: H’078>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R/W	R/W	R(0)/W(1)	R(0)/W(1)	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0: Buffer access is not possible. 1: Buffer access is possible.
14	INBUFM	0	R	Transmit buffer monitor When the specified pipe is in the sending direction, this bit indicates the FIFO buffer status of the specified pipe. 0: The FIFO buffer does not contain data that can be sent. 1: The FIFO buffer contains data that can be sent.
13 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10	ATREPM	0	R/W	Automatic response mode This bit prohibits or enables automatic response of the specified pipe. 0: Automatic response prohibited 1: Automatic response enabled. (A zero-length packet response is sent during transmission. For reception, an NAK response is sent and an NRDY interrupt is generated.)
9	ACLRM	0	R/W	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0: Prohibited 1: Enabled (all buffers are initialized)
8	SQCLR	0	R(0)/ W(1)	Toggle bit clear Specify 1 in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0: Writing disabled 1: DATA0 specified
7	SQSET	0	R(0)/ W(1)	Toggle bit set Specify 1 in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1. 0: Writing disabled 1: DATA1 specified
6	SQMON	0	R	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0: DATA0 1: DATA1

Bit	Bit Name	Initial Value	R/W	Description
5	PBUSY	0	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.
4 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1, 0	PID	All 0	R/W	Response PID These bits specify the response method for the next transaction in the specified pipe. 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response

(1) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the selected pipe. This bit is operated by this module.

The meaning of this bit varies as follows depending on the value of the values of the DIR, BFRE, and DCLRM bits:

Table 32B.14 BSTS Bit Operations

DIR bit Setting Value	BFRE bit Setting Value	DCLRM bit Setting Value	Meaning of the BSTS Bit
0	0	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
		1	Setting prohibited
	1	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible. This bit indicates 0 when 1 is written in the BCLF bit after reading data has finished.
		1	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
1	0	0	This bit indicates 1 when writing of send data in the FIFO buffer becomes possible, and indicates 0 when writing data has finished.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

(2) Transmit buffer monitor (INBUFM) bit

When the selected pipe is in the sending direction (DIR = 1), this module sets this bit when software (or the DMAC) has finished writing data to at least one FIFO buffer.

This module clears this bit when this module finishes sending all data from the FIFO buffer to which the data has been written. In double-buffer mode (DBLB = 1), this module clears this bit when this module has finished sending all data from the two FIFO buffers and software (or the DMAC) has not yet finished writing data to one FIFO buffer.

When the selected pipe is in the receiving direction (DIR = 0), this bit indicates the same value as that of the BSTS bit.

(3) Automatic response mode (ATREPM) bit

This bit can be set when the transfer type of the selected pipe is bulk transfer.

When this bit is 1, this module responds to tokens sent from the USB host as described below.

1. When the selected pipe is used for bulk IN transfer (TYPE = 01 and DIR = 1)
   When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an IN token by sending a zero-length packet.
   Each time this module receives ACK from the USB host (the sequence of one transaction is receiving an IN token, sending a zero-length packet, and then receiving ACK), this module updates (toggles) the sequence toggle bit (DATA- PID).
   This module does not generate BRDY and BEMP interrupts.

2. When the selected pipe is used for bulk OUT transfer (TYPE = 01 and DIR = 0)
   When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an OUT token (or a PING token) by sending an NAK response and generates an NRDY interrupt.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

To perform a USB communication with this bit set, the FIFO buffer must be empty. No data must be written to the FIFO buffer during the USB communication with this bit set.

When the transfer type of the selected pipe is isochronous transfer, this bit must always be 0.

(4) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.

Table 32B.15 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.

Table 32B.16 shows the cases that require this processing.

Table 32B.15 Buffer Contents This Core Clears When the ACLRM Bit is Set

No.	Contents to be Cleared by Setting the ACLRM Bit
(1)	Whole contents of the FIFO buffer allocated to the specified pipe (if the double buffer is set, both FIFO buffers are cleared)
(2)	If the transfer type of the specified pipe is Isochronous transfer, the interval count value is cleared.

Table 32B.16 Cases Requiring the ACLRM Bit to be Set

No.	Cases when Data Clear is Required
(1)	The whole contents of the FIFO buffer allocated to the specified pipe needs to be cleared.
(2)	The interval count value needs to be reset.
(3)	The value of the BFRE bit is changed.
(4)	The value of the DBLB bit is changed.
(5)	Forced termination of the transaction count function is performed.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(5) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.

Writing 1 to the SQCLR bit must be done when the PID of the selected pipe is NAK.

If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(6) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.

Writing 1 to the SQSET bit must be done when the PID of the selected pipe is NAK.

If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(7) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the selected pipe. This bit is operated by this module.

When the transfer type of the selected pipe is other than isochronous transfer, this module toggles this bit when a transaction ends normally. This module, however, does not toggle this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.

(8) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the selected pipe starts. This module changes this bit from 1 to 0 when the transaction ends normally.

Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(9) Response PID (PID) bits

These bits are used to specify, the type of response of this module for individual pipes.

The default of PID is NAK. When the selected pipe is used for USB transfers, the PID setting muse be changed to BUF. For the basic operations (without communication packet errors involved) of this module depending on the value of the PID bits, see **Table 32B.17**.

If you have changed the PID of a selected pipe from BUF to NAK while the selected pipe is performing a USB communication, check, that the PBUSY bit is 0 to confirm that the USB transfer through the selected pipe has actually changed to the NAK status. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

This module changes the value of the PID bits in the following cases:

1. When the selected pipe is in the receiving direction and set the SHTNAK bit for the selected pipe, this module sets the PID to NAK when this module recognizes the end of a transfer.
2. When this module has received a data packet of which the payload size is larger than the maximum packet size, this module sets the PID to STALL (PID = 11).
3. If this module detects a USB bus reset, this module sets the PID to NAK.

To change the PID from NAK (PID = 00) to STALL, write 10 to the PID bits. To change the PID from BUF (PID = 01) to STALL, write 11 to the PID bits.

To change the PID from STALL (PID = 11) to NAK, write 10 to the PID bits once, and then write 00 to the PID bits. To change the PID from STALL to BUF, change the PID to NAK once, and then change it to BUF.

Table 32B.17 Core Operations Depending on the PID Setting

PID bit Setting Value	Transfer Type (TYPE Bit Setting Value)	Transfer Direction (DIR Bit Setting Value)	Operation of This Core
00 (NAK)	Bulk ("TYPE = 01"), or Interrupt ("TYPE = 10")	Independent of the setting value	Sends a NAK response for a token from the USB host.
	Isochronous ("TYPE = 11")	Receiving direction ("DIR = 0")	Does not respond to a token from the USB host.
		Sending direction ("DIR = 1")	Sends a zero-length packet for a token from the USB host.
01 (BUF)	Bulk ("TYPE = 01")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK or NYET response. If receiving data is not possible, this module sends an NAK response. For a PING Token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Interrupt ("TYPE = 10")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Bulk ("TYPE = 01") or Interrupt ("TYPE = 10")	Sending direction ("DIR = 1")	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends an NAK response.
	Isochronous ("TYPE = 11")	Receiving direction ("DIR = 0")	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data. If receiving data is not possible, this module discards data.
		Sending direction ("DIR = 1")	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends a zero-length packet.
10 (STALL) or 11 (STALL)	Bulk ("TYPE = 01") or Interrupt ("TYPE = 10")	Independent of the setting value	Sends a STALL response for a token from the USB host.
	Isochronous ("TYPE = 11")	Independent of the setting value	Does not respond to a token from the USB host.

32B.2.16.2 PIPE6 Control Register [PIPE6CTR] <Address: H'07A>
PIPE7 Control Register [PIPE7CTR] <Address: H'07C>
PIPE8 Control Register [PIPE8CTR] <Address: H'07E>
PIPE9 Control Register [PIPE9CTR] <Address: H'080>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	—	—	—	—	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R(0)/W(1)	R(0)/W(1)	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0: Buffer access is not possible. 1: Buffer access is possible.
14 to 10	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
9	ACLRM	0	R/W	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0: Automatic buffer clear mode prohibited 1: Automatic buffer clear mode enabled (all buffers are initialized)
8	SQCLR	0	R(0)/W(1)	Toggle bit clear Specify 1 in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0: Disabled 1: DATA0 specified
7	SQSET	0	R(0)/W(1)	Toggle bit set Specify 1 in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1 0: Disabled 1: DATA1 specified
6	SQMON	0	R	Toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0: DATA0 1: DATA1
5	PBUSY	0	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0: The specified pipe is not used in the USB bus. 1: The specified pipe is being used in the USB bus.
4 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1, 0	PID	All 0	R/W	Response PID These bits specify the response method for the next transaction in the specified pipe 00: NAK response 01: BUF response (according to the buffer status) 10: STALL response 11: STALL response

(1) Buffer status (BSTS) bit

See **Section 32B.2.16.1(2), Transmit buffer monitor (INBUFM) bit**.

(2) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.

Table 32B.18 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.

Table 32B.19 shows the cases that require this processing.

Table 32B.18 Buffer Contents This Core Clears when the ACLRM Bit is Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	All contents of the FIFO buffer allocated to the selected pipe

Table 32B.19 Cases Requiring the ACLRM Bit to be Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	When clearing contents of the FIFO buffer allocated to the selected pipe
(2)	When resetting the interval counter
(3)	When the value of the BFRE bit is changed
(4)	When the transaction count function is terminated forcibly

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(3) Sequence toggle bit clear (SQCLR) bit

See **Section 32B.2.16.1(5), Sequence toggle bit clear (SQCLR) bit**.

(4) Sequence toggle bit set (SQSET) bit

See **Section 32B.2.16.1(6), Sequence toggle bit set (SQSET) bit**.

(5) Sequence toggle bit monitor (SQMON) bit

See **Section 32B.2.16.1(7), Sequence toggle bit monitor (SQMON) bit**.

(6) Pipe busy (PBUSY) bit

See **Section 32B.2.16.1(8), Pipe busy (PBUSY) bit**.

(7) Response PID (PID) bits

See **Section 32B.2.16.1(9), Response PID (PID) bits**.

32B.2.17 Transaction Counters (PIPExTRE)

32B.2.17.1 PIPE1 Transaction Counter Enable Register [PIPE1TRE] <Address: H'090>
PIPE2 Transaction Counter Enable Register [PIPE2TRE] <Address: H'094>
PIPE3 Transaction Counter Enable Register [PIPE3TRE] <Address: H'098>
PIPE4 Transaction Counter Enable Register [PIPE4TRE] <Address: H'09C>
PIPE5 Transaction Counter Enable Register [PIPE5TRE] <Address: H'0A0>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R/W	R(0)/W(1)	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
9	TRENB	0	R/W	Transaction counter enable This bit enables or disables the transaction counter. 0: Disables the transaction counter. 1: Enables the transaction counter.
8	TRCLR	0	R(0)/W(1)	Transaction counter clear This bit clears the transaction counter to 0. To clear the counter, write 1 to this bit. 0: Invalid 1: Clears the current transaction counter.
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

Note: Changing values of these register bits for a selected pipe must be done when the PID of the selected pipe is NAK.
If you change the value of a bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of the bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(1) Transaction counter enable (TRENB) bit

When software sets this bit for the selected pipe in the receiving direction after specifying a total number of packets in the TRNCNT bits, this module performs the following control when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits:

1. When the continuous transfer mode is used (CNTMD = 1), this module switches the FIFO buffer to the CPU side at the end of reception even if the FIFO buffer is not full.
2. When the SHTNAK bit is 1, this module changes the PID of the selected pipe to NAK when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits.
3. When the DENDE bit is 1 and the PKTMD bit is 0, this module asserts the DEND signal when reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.
4. When the BFRE bit is 1, this module asserts the BRDY interrupt signal when it finishes reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.

For the pipe in the sending direction, write 0 to this bit (TRENB bit). When not using the transaction count function, write 0 to this bit.

When using the transaction count function, specify a value in the TRNCNT bits before writing 1 to this bit. Also, write 1 to this bit before receiving the first packet among those to be counted by the transaction count function.

(2) Transaction counter clear (TRCLR) bit

When software sets this bit for a selected pipe, this module clears the current value of the transaction counter corresponding to the selected pipe, and then clears this bit.

32B.2.17.2 PIPE1 Transaction Counter Register [PIPE1TRN] <Address: H’092>
PIPE2 Transaction Counter Register [PIPE2TRN] <Address: H’096>
PIPE3 Transaction Counter Register [PIPE3TRN] <Address: H’09A>
PIPE4 Transaction Counter Register [PIPE4TRN] <Address: H’09E>
PIPE5 Transaction Counter Register [PIPE5TRN] <Address: H’0A2>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRNCNT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TRNCNT	All 0	R/W	When writing: Specifies the total number of packets to be received by the pertinent pipe (number of transactions). When reading: Indicates the specified number of transactions if TRENB is 0. Indicates the number of the currently counted transaction if TRENB is 1.

(1) Transaction counter (TRNCNT) bits

When software writes 1 to the TRENB bit after setting the total number of packets to be received for the selected pipe in the receiving direction, this module performs the control described in **Section 32B.2.17.1(1), Transaction counter enable (TRENB) bit**.

When the TRENB bit is 0, this module indicates, by these bits, the number of transactions set.

When the TRENB bit is 1, this module indicates, by these bits, the current number of transactions counted.

This module increments the value of the TRNCNT bits by 1 when the status of reception meets all the following conditions (a) to (c):

(a) The TRENB bit is 1.

(b) When a packet is received, the value of the TRCNT bits is not equal to “current count + 1.”

(c) The payload size of received packets has reached the value of the MXPS bits.

This module clears the TRNCNT bits to 0 when any of the following conditions (1) to (3) is met:

(1) All the following conditions (a) to (c) are met:

(a) The TRENB bit is 1.

(b) When a packet is received, the value of the TRCNT bits is equal to “current count + 1. ”

(c) The payload size of received packets has reached the value of the MXPS bits.

(2) Both of the following conditions (a) and (b) are met:

(a) The TRENB bit is 1.

(b) A short packet has been received.

(3) The following condition is met:

(a) Written 1 to the TRCLR bit.

For the pipe in the sending direction, write 0 to these bits (TRNCNT bits). When not using the transaction count function, write 0 to these bits.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the TRENB bit is 0.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

If you change the value of these bits, write 1 to the TRCLR bit before writing 1 to the TRENB bit.

32B.2.18 Low Power Control Register

32B.2.18.1 Low Power Control Register [LPCTRL] <Address: H'100>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HWUPM	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved Nothing is assigned to these bits. Fix these bits to 0.
7	HWUPM	0	R/W	0: Resumes the PHY from the low-power mode while the internal bus clock (P1ϕ) is operating. 1: Enables resume from the low-power mode while the internal bus clock (P1ϕ) is stopped.
6 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) HWUPM

This bit is used to specify whether to enable resumption from the low-power mode even while the internal bus clock (P1ϕ) is stopped.

0: Disables resumption while the internal bus clock (P1ϕ) is stopped.

1: Enables resumption while the internal bus clock (P1ϕ) is stopped.

This bit specifies whether to detect resume signaling while the internal bus clock (P1ϕ) is stopped. Whether to resume is controlled by the L1EXTMD bit. To resume from the low-power mode (LPM L1 state) while the internal bus clock (P1ϕ) is stopped, set both this bit and the L1EXTMD bit.

32B.2.19 Low Power Status Register

32B.2.19.1 Low Power Status Register [LPSTS] <Address: H’102>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SUSPM	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	SUSPM	0	R/W	USBPHY Suspend M control This bit controls the Suspend M signal to the USBPHY. 0: USBPHY suspend mode 1: USBPHY normal mode
13 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) USBPHY SuspendM control (SUSPM) bit

This bit is used to control the SuspendM signal to the USBPHY. By default, the value of this bit is 0 and the USBPHY is in suspend mode. To operate this module, write 1 to this bit.

When the SUSPM bit is 0 (that is, when the UTMI clock is stopped), data cannot be written to this module, but can only be read from this module. Note, however, that data can be written to the registers listed in **Table 32B.20**.

Table 32B.20 Registers That Allow Writing when the SUSPM Bit is 0

Address	Register Name
H’000	SYSCFG0
H’002	BUSWAIT
H’100	LPCTRL
H’102	SUSPMODE

Note that the values written to the SYSCFG0 register while the USBPHY clock is stopped (SUSPM = 0) will be applied after the USBPHY clock starts (SUSPM = 1).

When the L1EXTMD bit is 0, this bit (SUSPM bit) is controlled (set or cleared) by software. When L1EXTMD bit is 1, this bit is controlled by software for the transition to the L1 or L2 state, and controlled by hardware for resumption from the L1 or L2 state, regardless of the level (L1 or L2).

32B.2.20 PHY Function Control Register

32B.2.20.1 PHY Function Control Register [PHYFUNCTR] <Address: H’104>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SusMon	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	SusMon	0	R	This bit allows reading of the status of the Suspend M signal.
13 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) SusMon

SuspendM monitor bit (read only)

The status of the Suspend M signal can be read.

32B.2.21 PHY_OTG Control Register (PHYOTGCTR)

32B.2.21.1 PHY_OTG Control Register [PHYOTGCTR] <Address: H’10A>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DmPuDwn	DpPuDwn	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
10	DmPuDwn	1	R	Dm Pulldown monitor bit
9	DpPuDwn	1	R	Dp Pulldown monitor bit
8 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) DmPuDwn

DmPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DM side is disabled.

1: 15-kΩ Pulldown resistor control on the DM side is enabled.

(2) DpPuDwn

DpPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DP side is disabled.

1: 15-kΩ Pulldown resistor control on the DP side is enabled.

32B.2.22 Peripheral L1 Control Register 1 (PL1CTRL)

32B.2.22.1 Peripheral L1 Control Register 1 [PL1CTRL1] <Address: H'144>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	L1EXTMD	—	—	HIRDTHR[3:0]				DVSQ[3:0]				L1NEGOMD	L1RESPMD[1:0]		L1RESPEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
14	L1EXTMD	0	R/W	USBPHY control mode on resumption from the L1 state This bit controls the USBPHY resume operation on resumption from the L1 state. 0: Does not set the Suspend M bit when the Host K signal is received. 1: Sets the Suspend M bit when the Host K signal is received
13, 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11 to 8	HIRDTHR[3:0]	All 0	R/W	L1 response negotiation threshold HIRD threshold to be used for the L1NEGOMD bit The format is the same as the HIRD field of the HL1CTRL register.
7	DVSQ[3]	0	R	DVSQ extension bit This bit combined with the device state (DVSQ[2:0]) bit indicates the L1 state. 0000b: Powered state 0001b: Default state 0010b: Address state 0011b: Configured state 01xxb: Suspended state 10xxb: L1 state
6 to 4	DVSQ[2:0]	All 0	R	These bits mirror the DVSQ[2:0] bits in INTSTS0.
3	L1NEGOMD	0	R/W	L1 response negotiation control This bit sets the negotiation function using for the HIRD value. 0: Returns an ACK response if the received HIRD value is larger than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. 1: Returns an ACK response if the received HIRD value is smaller than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. This bit is valid only when the value of the L1RESPMD[1:0] bit is 11b.
2, 1	L1RESPMD[1:0]	All 0	R/W	L1 response mode These bits specify how to respond to an LPM token. 00b: NYET 01b: ACK 10b: STALL 11b: Response according to the value of the L1NEGOMD bit
0	L1RESPEN	0	R/W	L1 response enable This bit enables an L1 response. 0: Does not support LPM. 1: Supports LPM.

(1) L1 EXT mode (L1EXTMD) bit

This bit specifies how to control the SuspendM bit upon receiving the Host K signal when the USBPHY is stopped by setting the SuspendM bit in the L1 state.

0: Does not set the SuspendM bit when this module is resumed from the L1 state.

1: Sets the SuspendM bit when this module is resumed from the L1 state.

NOTES

1. The Host K period lasts a minimum of 50 µs. Therefore, the USBPHY might be unable to be resumed within the Host K period if the software settings for resume same as those for the suspend state are applied. Because the initial value of this bit is controlled by software, set this bit at initialization when the L1 state is to be supported.
2. For the transition to the L1 state, the SuspendM bit is controlled by software regardless of the value of this bit.
3. When this bit is set, the SuspendM bit will be set also at resumption from the L2 state.

(2) HIRD negotiation threshold (HIRDTHR[3:0]) bits

These bits specify the value of HIRD threshold to be used for the negotiation specified by the L1NEGOMD bit.

The format of the value is the same as that of the HIRD field in the HL1CTRL register.

(3) Device state extension (DVSQ[3]) bit

This bit is used as the fourth bit for the device state (DVSQ) bits.

0000b: Powered state

0001b: Default state

0010b: Address state

0011b: Configured state

01xxb: Suspended state

10xxb: L1 state

(4) Device status (DVSQ[2:0]) bits

These bits mirror the DVSQ[2:0] bits in the Interrupt Status Register (INTSTS0).

(5) L1 negotiation mode (L1NEGOMD) bit

This bit is used to specify the negotiation function using the HIRD value.

0: Returns an ACK response when the received HIRD value is larger than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.

1: Returns an ACK response when the received HIRD value is smaller than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.

This bit is valid only when the value of L1RESPMD[1:0] bits is 11b.

(6) L1 response mode (L1RSPMD[1:0]) bits

When the L1RSPED bit is set, this module responds to an LPM token according to the value of these bits. These bits specify how to respond to the LPM token.

00b: NYET

01b: ACK

10b: STALL

11b: Response according to the value of L1NEGOMD bit

(7) L1 response enable (L1RSPEN) bit

When this bit is 0, this module does not respond to the LPM token it receives. When this bit is 1, this module responds to the LPM token (it receives) according to the value of the LPMRESPMD[1:0] bits.

32B.2.23 Peripheral L1 Control Register 2

32B.2.23.1 Peripheral L1 Control Register 2 [PL1CTRL2] <Address: H'146>

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RWEMON	HIRDMON[3:0]				—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
12	RWEMON	0	R/W	This bit reflects the value of the RWE bit in the LPM token received last.
11 to 8	HIRDMON[3:0]	All 0	R/W	These bits reflect the value of the HIRD field in the LPM token received last.
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

(1) RWE value monitor (RWEMON) bit

This bit is referenced to monitor the value of the RWE bit in a received LPM token.

This bit reflects the value of the RWE bit in the LPM token received last.

(2) HIRD value monitor (HIRDMON) bits

These bits are referenced to monitor the value of the HIRD field in a received LPM token.

These bits reflect the value of the HIRD field in the LPM token received last.

32B.3 Next Register Set

32B.3.1 Next Source Address Register n

32B.3.1.1 Next0 Source Address Register ch0 [N0SA_0] <Address: H’400> Next1 Source Address Register ch0 [N1SA_0] <Address: H’40C> Next0 Source Address Register ch1 [N0SA_1] <Address: H’440> Next1 Source Address Register ch1 [N1SA_1] <Address: H’44C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA (in normal mode), WD (in write-only mode)															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA (in normal mode), WD (in write-only mode)															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA (in normal mode)	All 0	R/W	Source Address These bits specify the start address of the DMA transfer source.
	WD (in write-only mode)	All 0	R/W	Write Data These bits specify the write data in write-only mode.

Note: In a transfer in link mode, the data in the N0SA_n register is overwritten with descriptor read data.

32B.3.2 Next Destination Address Register n

32B.3.2.1 Next0 Destination Address Register ch0 [N0DA_0] <Address: H’404> Next1 Destination Address Register ch0 [N1DA_0] <Address: H’410> Next0 Destination Address Register ch1 [N0DA_1] <Address: H’444> Next1 Destination Address Register ch1 [N1DA_1] <Address: H’450>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA	All 0	R/W	Destination Address These bits specify the start address of the DMA transfer destination.

Note: In a transfer in link mode, the data in the N0DA_n register is overwritten with descriptor read data.

32B.3.3 Next Transaction Byte Register n

32B.3.3.1 Next0 Transaction Byte Register ch0 [N0TB_0] <Address: H’408> Next1 Transaction Byte Register ch0 [N1TB_0] <Address: H’414> Next0 Transaction Byte Register ch1 [N0TB_1] <Address: H’448> Next1 Transaction Byte Register ch1 [N1TB_1] <Address: H’454>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB	All 0	R/W	Transaction Byte These bits specify the total number of transfer bytes. *Note*: Do not start a DMA transaction with 0 set in these bits.

Note: The N0TB_n register is overwritten by the descriptor read data during link mode transfer.

32B.4 Current Register Set

32B.4.1 Current Source Address Register

32B.4.1.1 Current Source Address Register ch0 [CRSA_0] <Address: H'418> Current Source Address Register ch1 [CRSA_1] <Address: H'458>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRSA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA	All 0	R	Current Source Address Register This register indicates the read address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the SAD bit in the CHCFG_n register is 1, the value of this register is fixed. When the WONLY bit in the CHCFG_n register is 1, the value of this register is undefined.) The initial value of this register is loaded from the following register: In register mode: A transfer source address is loaded from the Next0/1 Register Set. In link mode: A transfer source address is loaded from the descriptor. (The descriptor read data is input to the N0SA_n register, and is loaded into the CRSA_n register when a transfer starts.) The value of this register is incremented when a read transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

32B.4.2 Current Destination Address Register

32B.4.2.1 Current Destination Address Register ch0 [CRDA_0] <Address: H’41C> Current Destination Address Register ch1 [CRDA_1] <Address: H’45C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRDA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA	All 0	R	Current Destination Address Register This register indicates the write address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the DAD bit in the CHCFG_n register is 1, the value of this register is fixed.) The initial value of this register is loaded from the following register: In register mode: A transfer destination address is loaded from the Next0/1 Register Set. In link mode: A transfer destination address is loaded from the descriptor. (The descriptor read data is input to the N0DA_n register, and is loaded into the CRDA_n register when a transfer starts.) The value of this register is incremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

32B.4.3 Current Transaction Byte Register

32B.4.3.1 Current Transaction Byte Register ch0 [CRTB_0] <Address: H’420> Current Transaction Byte Register ch1 [CRTB_1] <Address: H’460>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTB															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB	All 0	R	Current Transaction Byte Register This register indicates the remaining number of transfer bytes in the ongoing DMA transaction. The value of this register is decremented automatically while the DMA transaction is in process. The initial value of this register is loaded from the following register: In register mode: The number of transfer bytes is loaded from the Next0/1 Register Set. In link mode: The number of transfer bytes is loaded from the descriptor. (The descriptor read data is input to the N0TB_n register, and is loaded into the CRTB_n register when a transfer starts.) The value of this register is decremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

32B.5 Channel Register Set

32B.5.1 Channel Status Register n

32B.5.1.1 Channel Status Register ch0 [CHSTAT_0] <Address: H’424>
Channel Status Register ch1 [CHSTAT_1] <Address: H’464>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DNUM								—	—	—	—	—	SWPRQ	DMARQM	INTM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DNUM	All 0	R	Data Number These bits indicate the amount of valid data in the buffer. The indicated amount of data is the amount of the data that was read from the source but has not yet been written to the destination. (Unit: byte) Incrementing condition: ● A DMA read transfer ends. Decrementing condition: ● A DMA write transfer ends. Clearing conditions: ● A condition for clearing the EN bit is met. ● “1” is written to the SWRST bit in the CHCTRL_n register.
23 to 19	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.
18	SWPRQ	0	R	Sweep Request This bit indicates the state of sweep request. This bit indicates the state of software sweep request (which has been activated by the SETSSWPRQ bit in the CHCTRL_n register). 1: The sweep request signal has been asserted. 0: The sweep request signal has not been asserted. Setting condition: ● The SETSSWPRQ bit in the CHCTRL_n register is asserted. Clearing conditions: ● The buffer becomes empty because of sweep. ● “1” is written to the CLRHSWPRQM bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.

Bit	Bit Name	Initial Value	R/W	Description
17	DMARQM	0	R	DMAREQ Mask This bit indicates the state of temporary masking of the DMA transfer request from the USB control. 1: The request is temporarily masked. 0: The request is released from temporary masking. Setting condition: ● The SETDMARQM bit in the CHCTRL_n register is set. Clearing conditions: ● "1" is written to the CLRDMARQM bit in the CHCTRL_n register. ● "1" is written to the SWRST bit in the CHCTRL_n register.
16	INTM	0	R	Interrupt Mask This bit indicates the state of temporary masking of the output from the USBFDMAmn interrupt. 1: The output is temporarily masked. 0: The output is released from temporary masking. Setting condition: ● "1" is written to the SETINTM bit in the CHCTRL_n register. Clearing conditions: ● "1" is written to the CLRINTM bit in the CHCTRL_n register. ● "1" is written to the SWRST bit in the CHCTRL_n register.
15 to 12	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.
11	MODE	0	R	DMA Mode This bit indicates the DMA mode. The indicated value is the value of the DMS bit in the CHCFG_n register. 0: Register mode 1: Link mode
10	DER	0	R	Descriptor Error This bit indicates whether the data read from the descriptor is invalid (LV = 0) (regardless of the value of the DIM bit in the CHCFG_n register). 0: No descriptor error has occurred. 1: A descriptor error has occurred. Setting condition: ● In link mode, when the DRRP bit in the CHCFG_n register is 0, the LV bit value read from the descriptor is 0. Clearing conditions: ● "1" is written to the CLRDER bit in the CHCTRL_n register. ● "1" is written to the SWRST bit in the CHCTRL_n register.
9	DW	0	R	Descriptor WriteBack This bit indicates whether data is being written back to the descriptor. If a bus error occurs during write-back to the descriptor, this bit retains 1. 0: Status other than write-back to the header in link mode 1: (When the ER bit in the CHSTAT_n register is 0) Data is being written back to the header in link mode. (When the ER bit in the CHSTAT_n register is 1) A bus error has occurred during write-back to the header in link mode. Setting condition: ● Write-back to the header is started in link mode. Clearing conditions: ● Write-back to the header in link mode ends with an OK response. ● "1" is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1 because of an error response, this bit can be cleared only by setting the SWRST bit.

Bit	Bit Name	Initial Value	R/W	Description
8	DL	0	R	Descriptor Load This bit indicates whether data is being read from the descriptor. If a bus error occurs during descriptor reading, this bit retains 1. 0: Status other than descriptor reading 1: (When the ER bit is 0) Descriptor reading is in process in link mode. (When the ER bit is 1) A bus error has occurred during descriptor reading in link mode. Setting condition: ● Descriptor reading is started in link mode. Clearing conditions: ● Descriptor reading in link mode ends with an OK response. ● “1” is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1 because of an error response, this bit can be cleared only by setting the SWRST bit.
7	SR	0	R	Selected Register Set In register mode, this bit indicates the register set that is selected. 0: Next0 Register Set 1: Next1 Register Set Setting condition: ● The RSEL bit in the CHCFG_n register is set. Clearing condition: ● The RSEL bit in the CHCFG_n register is cleared.
6	TC	0	R	Terminal Count This status bit indicates whether the DMA transaction has ended. This bit is set only when the TCM bit in the CHCFG_n register is 0. 0: The DMA transfer has not ended. 1: The DMA transfer has ended. Setting conditions: ● In register mode, transfer of the total number of transfer bytes specified in the CRTB bits ends. ● In link mode, when the WBD bit in the header of the descriptor is 1, transfer of the total number of transfer bytes specified in the CRTB bits ends. ● In link mode, when the WBD bit in the header of the descriptor is 0, write-back to the descriptor ends. Clearing conditions: ● “1” is written to the CLRTC bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.
5	END	0	R	USBFDMAmn Interrupted This bit indicates whether the DMA transaction has ended and an USBFDMAmn interrupt has occurred. 0: The DMA transfer has not ended. 1: The DMA transfer has ended. Setting conditions: ● The condition for setting the TC bit is met, and the DEM bit in the CHCFG_n register is 0. ● In link mode, when the descriptor is read, the LV bit in the header is 0, and the DRRP and DIM bits in the CHCFG_n register are 0. Clearing conditions: ● “1” is written to the CLREND bit in the CHCTRL_n register. ● “1” is written to the SWRST bit in the CHCTRL_n register.

Bit	Bit Name	Initial Value	R/W	Description
4	ER	0	R	Error This bit indicates whether an error response has been received and a DMAERR interrupt has occurred during the DMA transfer. 0: No error response has been received. 1: An error response has been received. Setting condition: ● An error response is received in a bus cycle. Clearing condition: ● “1” is written to the SWRST bit in the CHCTRL_n register.
3	SUS	0	R	Suspend This bit indicates whether the channel is suspended. For details, see **Section 32B.9.13.8(2), Suspension**. 0: Channel_n is not suspended. 1: Channel_n is suspended. Setting condition: ● “1” is written to the SETSUS bit in the CHCTRL_n register during the DMA transfer using channel_n, and, thereby, the inside of the channel is suspended. Clearing conditions: ● “1” is written to the CLRSUS bit in the CHCTRL_n register. ● “1” is written to the CLREN bit in the CHCTRL_n register. ● A condition for clearing the EN bit in the CHSTAT_n register.
2	TACT	0	R	Transaction Active This bit indicates whether the DMAC is operating. This bit is used to check whether the channel is stopped fully. For details, see **Section 32B.9.13.8, Transfer state**. 0: The DMA in channel_n is stopped. 1: The DMA in channel_n is operating. Setting condition: ● “1” is written to the SETEN bit in the CHCTRL_n register (to start descriptor reading or wait for a DMA request). Clearing condition: ● The internal state is the idle state (the EN bit in the register has been cleared, and all transfers have ended).
1	RQST	0	R	Request This bit indicates whether a transfer request has been received. 0: No DMA transfer request has been received. 1: A DMA transfer request has been received. Setting conditions: ● “1” is written to the STG bit in the CHCTRL_n register. ● A DMA transfer request is received from the USB control. Clearing conditions: ● “1” is written to the SWRST bit in the CHCTRL_n register. ● “1” is written to the CLRRQ bit in the CHCTRL_n register. ● In single transfer mode (the TM bit in the CHCFG_n register is 0), a transfer is executed on the side specified by the REQD bit in the CHCFG_n register ● In register mode, all DMA transactions are complete (the REN bit in the CHCFG_n register is 0). ● In link mode, the DMA transfer of the last descriptor (LE = 1) ends. ● In link mode, a DMA transfer is stopped during descriptor reading (the LV bit is 0 and the DRRP bit in the CHCFG_n register is 0). ● In link mode, when the DEM bit in the CHCFG_n register is 0, a DMA transaction ends. ● The master interface receives a bus error signal.

Bit	Bit Name	Initial Value	R/W	Description
0	EN	0	R	Enable This bit indicates whether the operation of DMA channel n is enabled or stopped. 0: Operation is stopped. 1: Operation is enabled. Setting conditions: ● "1" is written to the SETEN bit in the CHCTRL_n register. Clearing conditions: ● "1" is written to the SWRST bit in the CHCTRL_n register. ● "1" is written to the CLREN bit in the CHCTRL_n register. ● An error response is received during transfer. ● In register mode, all DMA transactions are completed (the REN bit in the CHCFG_n register is 0). ● In link mode, DMA transfer of the last descriptor (the LE bit is 1) ends (if the WBD bit is 0, write- back to the descriptor ends). ● In link mode, reading of a descriptor is stopped (the LV bit is 0 and the DRRP bit in the CHCFG_n register is 0).

Note 1. When the ER bit in the CHSTAT_n register is set, treat the corresponding series of transfers as invalid transactions.

Note 2. To interrupt a DMA transaction, mask or clear transfer requests or clear the EN bit in the CHSTAT_n register. (For the procedure to interrupt, see **Section 32B.9.13.8(3), Transfer suspension**.)

Note 3. If the DMA transfer request from the USB control and the transfer request by software (setting the STG bit in the CHCFG_n register) are used together, the cause of activating the request that takes effect cannot be identified. Therefore, design the system so that only one type of transfer requests is used.

Note 4. When using the transfer request by software, operate the STG bit for a new transfer request only after the DMA transfer requested last ends (after checking the end of the last DMA transfer by referencing the Current Register Set or another method).

32B.5.2 Channel Control Register n

32B.5.2.1 Channel Control Register ch0 [CHCTRL_0] <Address: H’428> Channel Control Register ch1 [CHCTRL_1] <Address: H’468>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLRDMARQM	SETDMARQM	CLRINTM	SETINTM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SETSSWPRQ	—	SETREN	—	—	CLRSUS	SETSUS	CLRDER	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read
19	CLRDMARQM	0	R/W	Clear DMAREQ Mask Writing 1 to this bit releases the DMA transfer requests from the USB control from temporary masking. Writing 1 to this bit also clears the DMARQM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Releases the DNA transfer requests from masking set by using the SETDMARQM bit. 0: Has no effect on operation.
18	SETDMARQM	0	R/W	SET DMAREQ Mask Writing 1 to this bit temporarily masks the DMA transfer requests from the USB control. Writing 1 to this bit also sets the DMARQM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Masks the DMA transfer requests from the USB control. 0: Has no effect on operation.
17	CLRINTM	0	R/W	Clear Interrupt Mask Writing 1 to this bit releases the USBFDMAmn interrupt from masking. Writing 1 to this bit also clears the INTM bit in the CHSTATn register. Releasing the INT_DMA[n] pin output from masking when the LVINT bit in the DCTRL register and the END bit in the CHSTAT_n register are 1 activates the INT_DMA[n] pin output. (The pin output is not activated if the LVINT bit is 0.) When this bit is read, 0 is read. 1: Releases the pin output from masking set by using the SETINTM bit. 0: Has no effect on operation.
16	SETINTM	0	R/W	SETINTMSet Interrupt Mask Writing 1 to this bit temporarily masks the USBFDMAmn interrupt. Writing 1 to this bit also sets the INTM bit in the CHSTATn register. When this bit is read, 0 is read. 1: Masks the USBFDMAmn interrupt. 0: Has no effect on operation.
15	—	0	R	Reserved area. Write 0 to this bit. When this bit is read, 0 is read.

Bit	Bit Name	Initial Value	R/W	Description
14	SETSSWPRQ	0	R/W	Set Software Sweep Request Writing 1 to this bit sweeps out the data stored in the buffer to the destination (see **Section 32B.9.13.3(1), Forced software sweeping request**). When this bit is read, 0 is read. 1: Writes, to the destination, the data that is stored in the buffer and has not yet been written to the destination. 0: Has no effect on operation. If the destination asserts a hardware request (REQD = 1), the sweep operation cannot be used.
13	—	0	R	Reserved area. Write 0 to this bit. When this bit is read, 0 is read.
12	SETREN	0	R/W	Set Register Set Enable Writing 1 to this bit sets the REN bit in the CHCFG_n register. When this bit is read, 0 is read. 1: Sets the REN bit in the CHCFG_n register. 0: Has no effect on operation.
11, 10	—	All 0	R	Reserved area. Write 0 to these bits. When these bits are read, 0 is read.
9	CLRSUS	0	R/W	Clear Suspend Writing 1 to this bit when the SUS bit in the CHSTAT_n register is 1 releases the ongoing DMA transfer from the suspended state. When this bit is read, 0 is read. 1: Releases the ongoing DMA transfer from the suspended state. 0: Has no effect on operation.
8	SETSUS	0	R/W	Set Suspend Writing 1 to this bit when the EN bit in the CHSTAT_n register is 1 suspends the ongoing DMA transfer. When this bit is read, 0 is read. 1: Suspends the ongoing DMA transfer. 0: Has no effect on operation.
7	CLRDER	0	R/W	Clear DER Writing 1 to this bit clears the DER bit in the CHSTAT_n register. Writing 1 to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0 is read. 1: Clears the DER bit. 0: Has no effect on operation.
6	CLRTC	0	R/W	Clear TC Writing 1 to this bit clears the TC bit in the CHSTAT_n register. When this bit is read, 0 is read. 1: Clears the TC bit. 0: Has no effect on operation.
5	CLREND	0	R/W	Clear End Writing 1 to this bit clears the END bit in the CHSTAT_n register. Writing 1 to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0 is read. 1: Clears the END bit. 0: Has no effect on operation.
4	CLRRQ	0	R/W	Clear Request Writing 1 to this bit clears the RQST bit in the CHSTAT_n register. When this bit is read, 0 is read. 1: Clears the RQST bit in the CHSTAT_n register. 0: Has no effect on operation.

Bit	Bit Name	Initial Value	R/W	Description
3	SWRST	0	R/W	Software Reset Writing 1 to this bit clears individual bits in the CHSTAT_n register (for the bits to be cleared, see the description of each bit). Setting this bit must be done when the EN and TACT bits are 0. When this bit is read, 0 is read. 1: Clears individual bits in the CHSTAT_n register. 0: Has no effect on operation.
2	STG	0	R/W	Software Trigger Writing 1 to this bit makes software set an internal transfer request. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority. When this bit is read, 0 is read. 1: Makes software set a transfer request (set the RQST bit in the CHSTAT_n register). 0: Has no effect on operation.
1	CLREN	0	R/W	Clear Enable Writing 1 to this bit clears the EN bit in the CHSTAT_n register (for details, see **Section 32B.9.13.8(3) Transfer suspension**). When this bit is read, 0 is read. 1: Disables DMA transfers (clears the EN bit in the CHSTAT_n register). 0: Has no effect on operation.
0	SETEN	0	R/W	Set Enable Writing 1 to this bit enables DMA transfers in DMA channel n. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority, and DMA transfers do not start. When this bit is read, 0 is read. 1: Enables DMA transfers (sets the EN bit in the CHSTAT_n register). 0: Has no effect on operation.

Note: Temporary masking (using the CLRDMARQM, and SETDMARQM bits) of the DMA transfer requests from the USB control applies to only the resources for channel n. Setting the SETDMARQM bit for channel n does not affect the operation of channel m.

32B.5.3 Channel Configuration Register

32B.5.3.1 Channel Configuration Register ch0 [CHCFG_0] <Address: H’42C> Channel Configuration Register ch1 [CHCFG_1] <Address: H’46C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	DIM	TCM	DEM	WONLY	—	DAD	SAD	DDS			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS				DRRP	—	—	—	—	—	—	—	REQD	—	—	SEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0	R/W	DMA Mode Select This bit specifies the DMA mode to be used. 0: Register mode (default) 1: Link mode
30	REN	0	R/W	Register Set Enable This bit specifies whether to successively perform, after a DMA transaction ends, another DMA transaction using the Next Register Set selected by the RSEL bit. This bit is valid only in register mode. 0: Does not perform the DMA transaction successively. 1: Performs the DMA transaction successively. Setting conditions: ● “1” is written to this bit. ● “1” is written to the SETREN bit in the CHCTRL_n register. Clearing conditions: ● “0” is written to this bit. ● The REN bit is 1, and a DMA transaction ends. To re-set the REN bit during a transaction, we recommend you to use the SETREN bit in the CHCTRL_n register.
29	RSW	0	R/W	Register Select Switch This bit specifies whether to automatically invert the RSEL bit after a DMA transaction ends. This bit is valid only in register mode. 0: Does not invert the RSEL bit after a DMA transaction ends. (Default) 1: Inverts the RSEL bit after a DMA transaction ends.
28	RSEL	0	R/W	Register Set Select This bit is used to select the Next Register Set to be used for the next DMA transaction. This bit is valid only in register mode. When the RSW bit is 1, this bit is inverted automatically at the end of a DMA transaction. 0: Uses the Next0 Register Set. (Default) 1: Uses the Next1 Register Set. Transition condition: A DMA transaction ends with the RSW bit set.

Bit	Bit Name	Initial Value	R/W	Description
27	SBE	0	R/W	Sweep Buffer Enable This bit specifies whether to sweep (write) the data already read and stored in the buffer and stop transfer when the EN bit in the CHSTAT_n register is cleared during a DMA transaction. The sweep mode can be used only when the REQD bit is 0. 0: Stops transfer without sweeping out the buffer. (Default) 1: Stops transfer after sweeping out the buffer.
26	DIM	0	R/W	Descriptor Interrupt Mask This bit specifies whether to mask the USBFDMAmn interrupt when the LV bit value read from descriptor header is 0. 0: Does not mask the USBFDMAmn interrupt. (Default) 1: Mask the USBFDMAmn interrupt.
25	TCM	0	R/W	DMATC Mask This bit is used to mask the DMATC signal, which is sent from the DMAC to USB control. When this bit is 1 at the time the DMATC signal is to be output, the DMATC signal is not asserted. Also, the TC bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is automatically cleared. In link mode, this bit is not cleared automatically. Use this bit when you control DMA transfers by software. 0: Does not mask the DMATC signal. (Default) 1: Masks the DMATC signal. Clearing condition: A DMA transaction ends with the TCM bit set.
24	DEM	0	R/W	USBFDMAmn Mask When this bit is 1 at the time the USBFDMAmn interrupt is not asserted. Also, the END bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is not cleared automatically. In link mode, this bit is automatically cleared. 0: Does not mask the USBFDMAmn interrupt. (Default) 1: Masks the USBFDMAmn interrupt. Clearing condition: A DMA transaction ends with the DEM bit set.
23	WONLY	0	R/W	Write Only Mode This bit is used to switch the transfer operation mode to the write-only mode (see **Section 32B.9.12.2, Write only mode**). 0: Normal operation (default) 1: Write-only mode
22	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
21	DAD	0	R/W	Destination Address Direction This bit specifies the direction of counting the transfer-destination address in DMA channel n. If the transfer destination is on the USB control side, write 1 (fixed) to this bit. 0: Incrementing (default) 1: Fixed When the transfer destination uses the skip mode or is beat-unaligned, do not specify 1 (fixed) in this bit.
20	SAD	0	R/W	Source Address Direction This bit specifies the direction of counting the transfer-source address in DMA channel n. If the transfer source is on the USB control side, write 1 (fixed) to this bit. 0: Incrementing (default) 1: Fixed When the transfer source uses the skip mode or is beat-unaligned, do not specify 1 (fixed) in this bit.

Bit	Bit Name	Initial Value	R/W	Description
19 to 16	DDS[3:0]	All 0	R/W	Destination Data Size These bits specify the size of DMA transfer data. When the transfer destination is on the USB control side, select the normal mode. Use bit 3 to switch between the normal and skip modes. 0: Normal mode (default) 1: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see **Table 32B.21**.) 000: 8 bits (default) 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits 101: 256 bits 110: 512 bits 111: Setting prohibited
15 to 12	SDS[3:0]	All 0	R/W	Source Data Size These bits specify the size of DMA transfer data. Use bit 3 to switch between the normal and skip modes. 0: Normal mode (default) 1: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see **Table 32B.21**.) 000: 8 bits (default) 001: 16 bits 010: 32 bits 011: 64 bits 100: 128 bits 101: 256 bits 110: 512 bits 111: Setting prohibited
11	DRRP	0	R/W	Descriptor Read Repeat This bit switches the operation to be performed when the LV value in the header read from the descriptor is 0. (See **section 32B.9.12.1(2)(a), Operation flow of link mode**.) • 0: This module sets the DER bit in the CHSTAT_n register, and then stops descriptor reading. (Default) • 1: This module keeps reading the same descriptor until the LV value changes to 1, and, when the LV value becomes 1, starts the DMA transfer using the values in the descriptor. The interval of descriptor reading is controlled by using the DSCITVL register.
10 to 7	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
6, 5	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 1.
4	—	0	R	Nothing is assigned to this bit. Fix this bit to 0.
3	REQD	0	R/W	Request Direction This bit specifies whether the USB control is on the transfer source side or it is on the transfer destination side. 0: The USB control is on the transfer source side. (Default) 1: The USB control is on the transfer destination side.
2, 1	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
0	SEL	0	R/W	Terminal Select This bit selects the FIFO channel to be used on the USB control side. 0: D0FIFO 1: D1FIFO

The range of values specifiable in the SDS[2:0] and DDS[2:0] bits depends on the data bus width, number of implemented buffer stages, and whether the address to be accessed is beat-aligned or not (beat-unaligned). The table below shows the range of specifiable values.

Table 32B.21 Range of Sizes That Can Be Specified in SDS and DDS Bits

Transfer Address	REQD	SDS[2:0]	DDS[2:0]
Beat aligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)
	1	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.
Beat unaligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)
	1	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.

Note: When the destination is beat-unaligned with REQD = 0 or the source is beat-unaligned with REQD = 1, specify values in the range of specifiable values for beat-unaligned transfer addresses in both of the SDS[2:0] and DDS[2:0] bits.

Even if the transfer source and destination are beat-aligned when a DMA transaction starts, they might become beat-unaligned in the middle of the transaction when a skip transfer is used. If this might occur, perform settings in the first place on the assumption that the transfer source and destination are beat-unaligned.

If software cannot determine whether the transfer source and/or destination is beat-aligned, use values in the range of specifiable values for beat-unaligned transfer addresses.

32B.5.4 Channel Interval Register n

For details, see **Section 32B.9.13.6, Interval Count Function**.

32B.5.4.1 Channel Interval Register ch0 [CHITVL_0] <Address: H’430> Channel Interval Register ch1 [CHITVL_1] <Address: H’470>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ITVL															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
15 to 0	ITVL	All 0	R/W	Interval These bits specify the DMA transfer interval.

32B.5.5 Channel Extension Register n

32B.5.5.1 Channel Extension Register ch0 [CHEXT_0] <Address: H'434> Channel Extension Register ch1 [CHEXT_1] <Address: H'474>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DPR				—	—	—	—	SPR			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
11 to 8	DPR	All 0	R/W	Destination PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA write transfer. The initial value of these bits is H'0.
7 to 4	—	All 0	R/W	Nothing is assigned to these bits. Fix these bits to 0.
3 to 0	SPR	All 0	R/W	Source PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA read transfer. The initial value of these bits is H'0.

32B.6 Link Register Set

When software sets a descriptor address in the NXLA_n register and starts the DMAC, hardware loads the value set in the NXLA_n register into the CRLA_n register. Then, the descriptor is read, and the DMAC starts a DMA transaction according to the values read from the descriptor. The value in the NXLA_n register is automatically updated to the Next Link Address value read from the descriptor, and the updated value is used as the descriptor address in the next DMA transaction.

32B.6.1 Next Link Address Register n (NXLA_n)

32B.6.1.1 Next Link Address Register ch0 [NXLA_0] <Address: H’438> Next Link Address Register ch1 [NXLA_1] <Address: H’478>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	NXLA	All 0	R/W	Next Link Address These bits specify the link-destination address. Upper two bits are fixed to 0, and only the address aligned with a word boundary can be set.
1, 0		All 0	R	

32B.6.2 Current Link Address Register n (CRLA_n)

32B.6.2.1 Current Link Address Register ch0 [CRLA_0] <Address: H’43C> Current Link Address Register ch1 [CRLA_1] <Address: H’47C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRLA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA	All 0	R	Current Link Address These bits indicate the address of the descriptor being used for current transaction.

32B.7 Skip Register Set

This register set is used to specify settings for a skip (scatter/gather) transfer.

32B.7.1 Source Continuous Register n

32B.7.1.1 Source Continuous Register ch0 [SCNT_0] <Address: H'600> Source Continuous Register ch1 [SCNT_1] <Address: H'620>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SCNT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCNT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SCNT	All 0	R/W	Source Continuous These bits specify the size of the space to be accessed continuously by source address access. (Unit: byte)

Note: This register is used in pair with the SSKP_n register (see **Figure 32B.3**). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not be set to 1 (Fixed). Do not perform a skip transfer with the SCNT bits set to 0.

32B.7.2 Source Skip Register n

32B.7.2.1 Source Skip Register ch0 [SSKP_0] <Address: H’604> Source Skip Register ch1 [SSKP_1] <Address: H’624>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSKP															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSKP															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SSKP	All 0	R/W	Source Skip These bits specify the size of area to be skipped in a source address access. (Unit: byte)

Note: This register is used in pair with the SCNT_n register (see **Figure 32B.3**). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not set to 1 (Fixed).

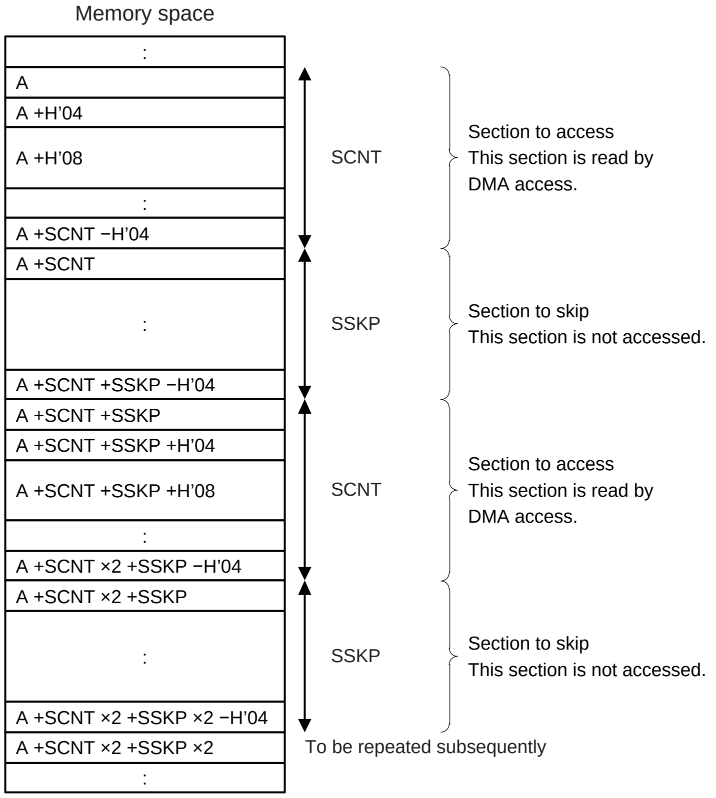


Figure 32B.3 Relationship between SSKP and SCNT

You can specify values of the SCNT and SSKP bits regardless of the source address and the value of the SDS field in the CHCFG_n register. The DMAC performs access based on the size specified in the SDS field, and fetches only valid data into the buffer (see **Section 32B.9.14.1(1), Read access**).

32B.7.3 Destination Continuous Register n

32B.7.3.1 Destination Continuous Register ch0 [DCNT_0] <Address: H’608> Destination Continuous Register ch1 [DCNT_1] <Address: H’628>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCNT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCNT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DCNT	All 0	R/W	Destination Continuous These bits specify the size of the space to be accessed continuously by destination address access. (Unit: byte)

Note: This register is used in pair with the DSKP_n register (see **Figure 32B.4**). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1 (Fixed). Do not perform a skip transfer with the DCNT bits set to 0.

32B.7.4 Destination Skip Register n

32B.7.4.1 Destination Skip Register ch0 [DSKP_0] <Address: H’60C> Destination Skip Register ch1 [DSKP_1] <Address: H’62C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSKP															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSKP															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSKP	All 0	R/W	Destination Skip These bits specify the size of area to be skipped in a destination address access. (Unit: byte)

Note: This register is used in pair with the DCNT_n register (see **Figure 32B.4**). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1 (Fixed).

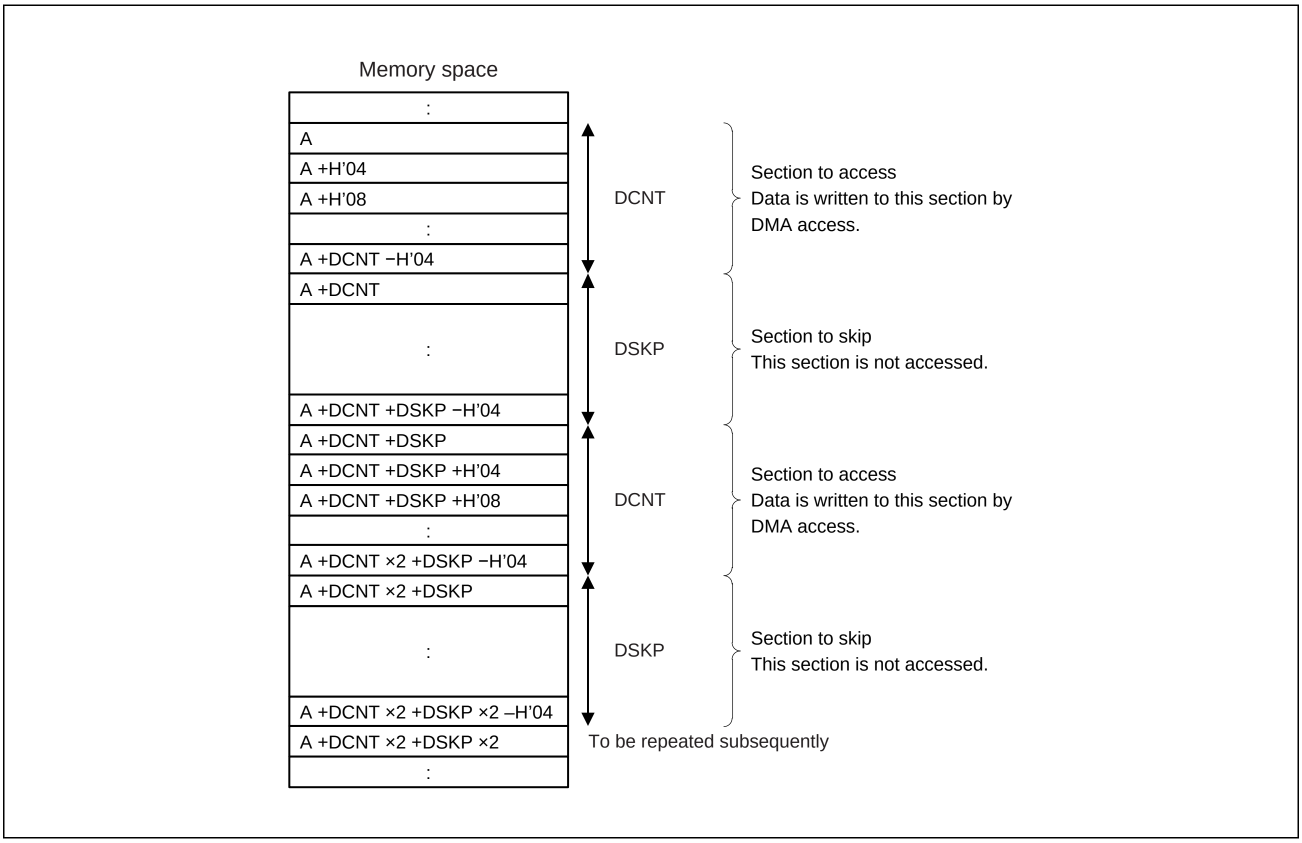


Figure 32B.4 Relationship between DSKP and DCNT

You can specify values of the DCNT and DSKP bits regardless of the destination address and the value of the DDS field in the CHCFG_n register. The DMAC performs write access to only the specified space that has a combined size not more than the size specified in the DDS field (see **Section 32B.9.14.1(1), Read access**).

32B.8 DMA Register Set

The registers described below are shared by all channels.

32B.8.1 DMA Control Register

32B.8.1.1 DMA Control Register [DCTRL] <Address: H’700>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	LWPR				—	—	—	—	LDPR			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LVINT	PR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
27 to 24	LWPR	All 0	R/W	Link WriteBack PROT These bits specify the value to be output to the MHPROT[3:0] pin at write-back to the descriptor in link mode.
23 to 20	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
19 to 16	LDPR	All 0	R/W	Link Descriptor PROT These bits specify the value to be output to the MHPROT[3:0] pin at reading of the descriptor in link mode.
15 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	LVINT	0	R/W	Level Interrupt To use this module, be sure to set this bit to 1.
0	PR	0	R/W	Priority This bit specifies the transfer priority control mode (see **Section 32B.9.13.2, DMA channel priority control**). 0: Fixed priority mode 1: Round-robin mode

32B.8.2 Descriptor Interval Register n

32B.8.2.1 Descriptor Interval Register [DSCITVL] <Address: H'704>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DITVL								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
15 to 8	DITVL	All 0	R/W	Descriptor Interval These bits specify the interval of descriptor read operation. The descriptor will be re-read at intervals of "value of DITVL x 256."
7 to 0	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.

32B.8.3 DMA Control Register

32B.8.3.1 DMA Status EN Register [DSTAT_EN] <Address: H'710>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EN1	EN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	EN1	0	R	This bit indicates the state of the EN bit for DMA channel 1.
0	EN0	0	R	This bit indicates the state of the EN bit for DMA channel 0.

32B.8.4 DMA Status ER Register

32B.8.4.1 DMA Status ER Register [DSTAT_ER] <Address: H’714>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ER1	ER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	ER1	0	R	This bit indicates the state of the ER bit for DMA channel 1.
0	ER0	0	R	This bit indicates the state of the ER bit for DMA channel 0.

32B.8.5 DMA Status END Register

32B.8.5.1 DMA Status END Register [DSTAT_END] <Address: H’718>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	END1	END0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	END1	0	R	This bit indicates the state of the END bit for DMA channel 1.
0	END0	0	R	This bit indicates the state of the END bit for DMA channel 0.

32B.8.6 DMA Status TC Register

32B.8.6.1 DMA Status TC Register [DSTAT_TC] <Address: H’71C>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TC1	TC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	TC1	0	R	This bit indicates the state of the TC bit for DMA channel 1.
0	TC0	0	R	This bit indicates the state of the TC bit for DMA channel 0.

32B.8.7 DMA Status SUS Register

32B.8.7.1 DMA Status SUS Register [DSTAT_SUS] <Address: H’720>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SUS1	SUS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Nothing is assigned to these bits. Fix these bits to 0.
1	SUS1	0	R	This bit indicates the state of the SUS bit for DMA channel 1.
0	SUS0	0	R	This bit indicates the state of the SUS bit for DMA channel 0.

32B.9 Functions

32B.9.1 System Control and Oscillation Control

This chapter describes register manipulations required to perform initial setup of this module. This chapter also describes the registers required to control power consumption.

For the parts of the sequence that are required in both host and peripheral modes, see **Section 32A.9.1, Host/Peripheral Common Setting Sequence**.

32B.9.1.1 USB data bus resistor control

This module controls switchover between the pull-up resistors for the D+ signal and the pull-down resistors for the D-signal of the USBPHY. Use the DPRPU bit and DRPD bit in the SYSCFG0 register to set pull-up or pull-down of each signal.

Recognize that a connection to the USB host is established, and then set 1 for the DPRPU bit in the SYSCFG0 register to pull up the D+ signal.

After connected to the host, this module automatically switches the resistor when the state changes to reset handshake, suspend, or resume.

If 0 is set for the DPRPU bit in the SYSCFG0 register during communication with the host, the pull-up resistor (or termination resistor) for the USB data line is disabled. This can notify the host controller of disconnection of a device.

32B.9.2 Interrupt Function

32B.9.2.1 Overview of Interrupt Function (other than DMA Master)

The following shows a list of interrupt functions of this module. An interrupt is notified as U2P_IXL_INT. Check the status register to identify the interrupt factor. The interrupt from this module is not asserted when the supply of the internal bus clock (P1ϕ) is stopped.

Table 32B.22 List of Interrupt functions

Bit	Interrupt name	Interrupt factor	Related status
VBINT	VBUS interrupt	The status change of the VBUS input pin is detected. (Changes from L to H and from H to L are detected.)	VBSTS
RESM	Resume interrupt	In the suspended state, a change of the USB bus status is detected (from J-State to K-State or from J-State to SE0).	—
SOFR	Frame number update interrupt	If SOFRM is 0: An SOF packet with a different frame number is received. If SOFRM is 1: An SOF with μframe number 0 cannot be received due to a problem such as packet corruption.	—
DVST	Device state transition interrupt	A transition of a device state is detected. USB bus reset detected Suspended state detected Set Address request received Set Configuration request received	DVSQ
CTRT	Control transfer stage transition interrupt	A transition of a control transfer stage is detected. Setup stage completed ControlWrite transfer status stage transition ControlRead transfer status stage transition Control transfer completed Control transfer sequence error	CTSQ
BEMP	Buffer empty interrupt	All data in the buffer memory is sent and the buffer becomes empty. A packet exceeding the maximum packet size is received.	PIPEBEMP
NRDY	Buffer not ready interrupt	A token is received when the PID setting is BUF and the buffer memory is not available for sending and receiving data. A CRC error or bit stuff error occurs when data is received in isochronous transfer. An interval error occurs when data is received in isochronous transfer.	PIPENRDY
BRDY	Buffer ready interrupt	The buffer becomes ready (available for reading or writing data).	PIPEBRDY

The following shows the relationship between interrupts of this module.

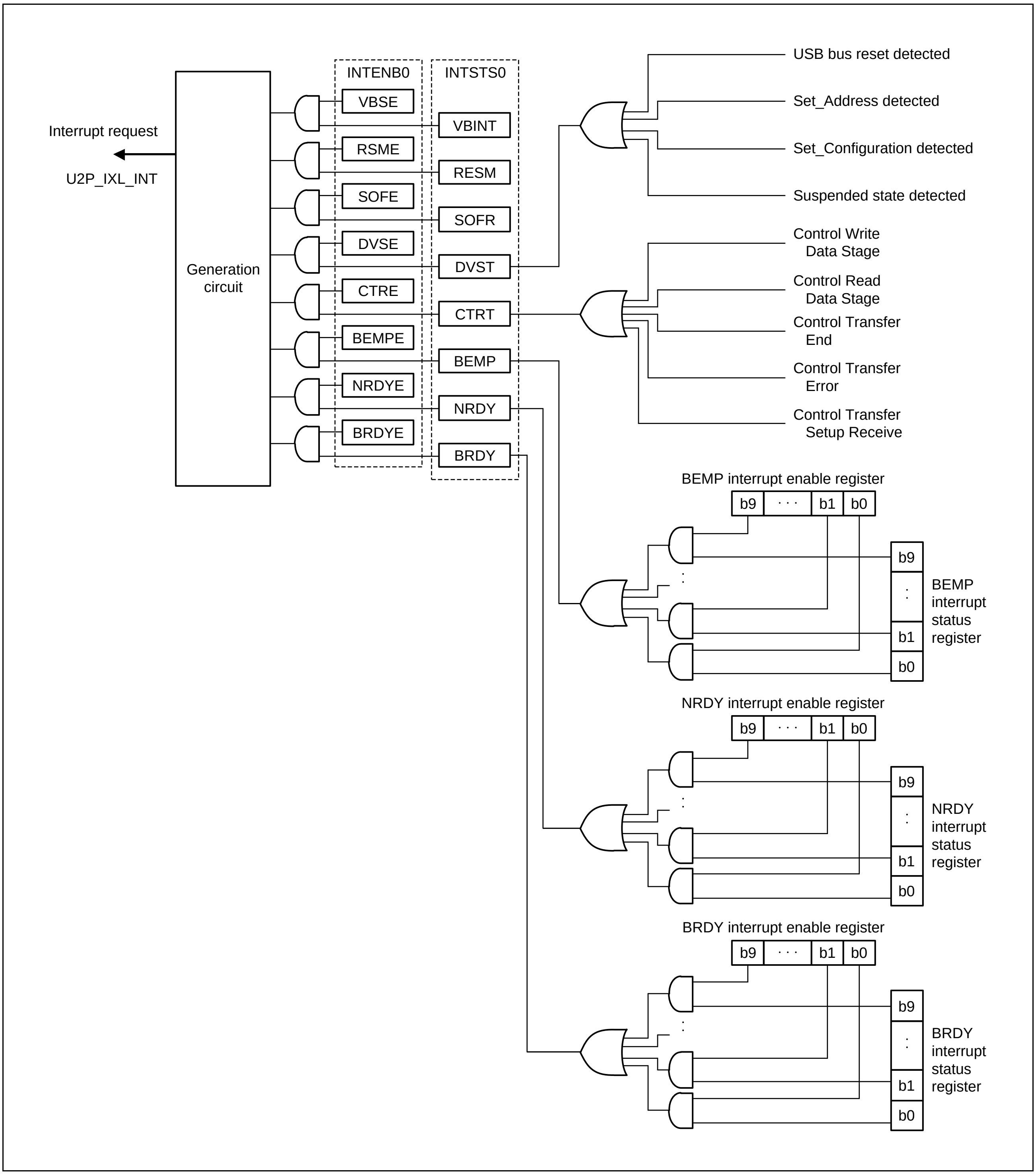


Figure 32B.5 Interrupt Association Diagram

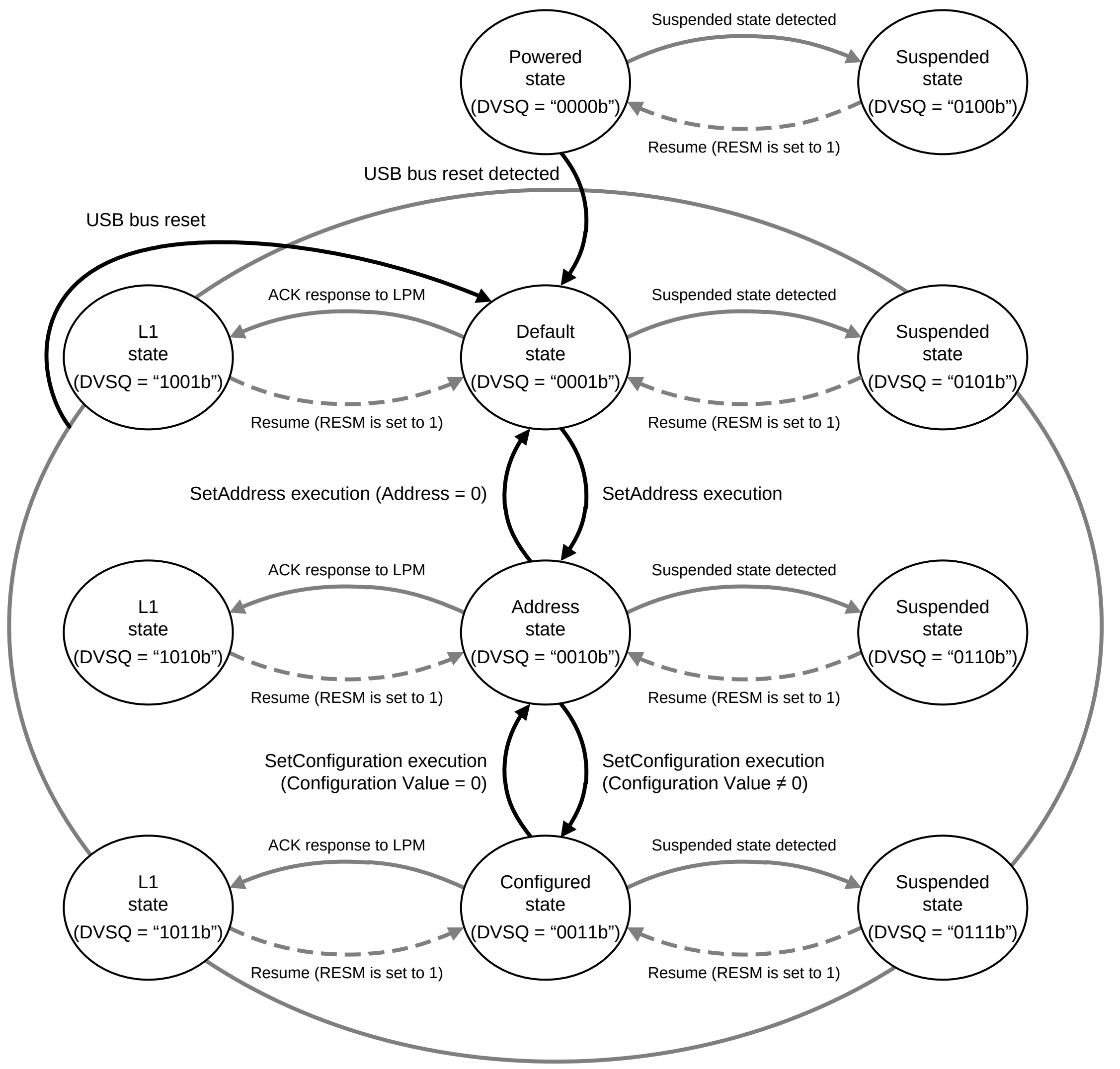
32B.9.2.2 Device State Transition Interrupts

Figure 32B.6 shows the device state transition diagram of this module.

This module manages device states and generates device state transition interrupts. However, resumption from the suspended state (resume signal detection) is detected by a resume interrupt.

Device state transition interrupts can be enabled and prohibited by using the INTENB0 register. A device state for which a transition has occurred can be checked in the DVSQ bits in the INTSTS0 register.

To trigger a transition to the default state, a device state transition interrupt is generated after a reset hand-shake protocol ends.



Note: When a transition indicated by a solid line occurs, the DVST bits are set to 1.
When a transition indicated by a broken line occurs, the RESM bit is set to 1.

Figure 32B.6 Device State Transition Diagram

32B.9.2.3 Control Transfer Stage Transition Interrupts

Figure 32B.7 shows the control transfer stage transition diagram of this module. This module manages a sequence of control transfers, and generates a control transfer stage transition interrupt. Control transfer stage transition interrupts can be enabled and prohibited by using the INTENB0 register. A transfer stage for which a transition has occurred can be checked in the CTSQ bits in the INTSTS0 register.

The following describes sequence errors that can occur during control transfer. If an error occurs, the PID bits in the DCPCTR register are set to 1X (STALL).

(1) For Control Read transfer

(a) An OUT or PING token is received in a situation where data has not been transferred yet for an IN token of the data stage.

(b) An IN token is received in the status stage.

(c) A data packet "DATAPID = DATA0" is received in the status stage.

(2) For Control Write transfer

(a) An IN token is received in a situation where an ACK response has not been sent yet for an OUT token of the data stage.

(b) The first data packet ”DATAPID = DATA0” is received in the data stage.

(c) An OUT or PING token is received in the status stage.

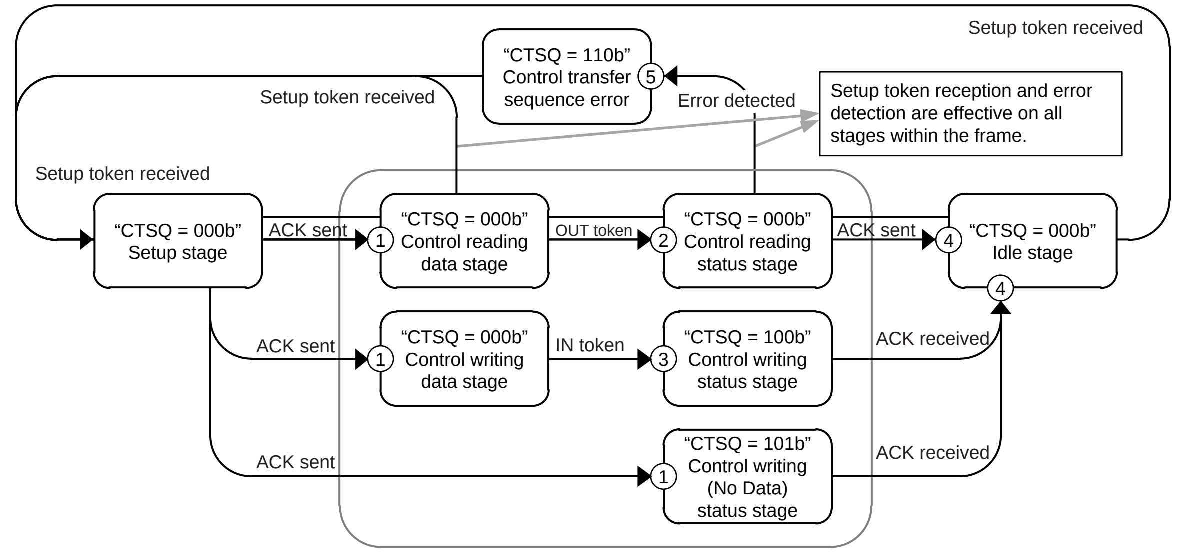
(3) For Nodata Control transfer

(a) An OUT or PING token is received in the status stage.

If the amount of received data exceeds the value of the wLength field in a USB request in the Control Write transfer data stage, this module cannot identify this situation as a control transfer sequence error. If a packet other than a zero-length packet is received in the Control Read transfer status stage, this module sends an ACK response and then terminates processing normally.

If a CTRT interrupt is generated (SERR bit is set to 1) due to a sequence error, the CTSQ bits retain 110 until the system writes 0 to the CTRT bits to clear the interrupt status.

Therefore, as long as the CTSQ bits retain 110, even if a new USB request is received, a CTRT interrupt that reports completion of a setup stage is not generated. (Information on completion of the setup stage is retained by this module, and a CTRT interrupt is generated after the interrupt status is cleared.)



Note: CTRT interrupt

① Setup stage completed

② ControlRead transfer status stage transition

③ ControlWrite transfer status stage transition

④ Control transfer completed

⑤ Control transfer sequence error

Figure 32B.7 Control Transfer Stage Transition Diagram

32B.9.2.4 Interrupts Relating to DMA Master

Table 32B.23 List of Interrupts

Interrupt Name	Interrupt Type
USBFDMAmn (m, n = 0, 1)	A DMA transaction ends.
	An invalid descriptor is read in link mode.
USBFDMAERRm (m = 0, 1)	An error response is returned for a transfer issued by the master interface.

This module has two types of DMA interrupts USBFDMAmn and USBFDMAERRm.

(1) USBFDMAmn interrupt (m, n = 0, 1)
This interrupt occurs when a DMA transaction ends or when an invalid descriptor is read in link mode.
An interrupt is divided for each DMA channel. USBFDMAm0 corresponds to the interrupt for DMA 0ch, and USBFDMAm1 corresponds to the interrupt for DMA 1ch.

(2) USBFDMAERRm interrupt
This interrupt occurs when an error response is returned for a transfer issued by the master interface. This interrupt applies to all DMA channels.

USBFDMAmn interrupt output can be temporarily masked by setting a register.

Interrupt detection can also be masked by setting the register. If interrupt detection is masked, the status register that indicates generation of an interrupt does not change.

On the other hand, a USBFDMAERRm interrupt signal does not have the masking function.

32B.9.3 Pipe Control

Table 32B.24 lists the pipe settings for this module. For USB data transfer, logical pipes called endpoints are used to enable data communication. This module provides 16 pipes for data transfer. Set each pipe according to the specification of the system.

Table 32B.24 PIPE Settings

Register Name	Bit Name	Setting	Comment
DCPCFG	TYPE	Specifies the transfer type.	Pipes 1 to 9: Settable
PIPECFG	BFRE	Selects BRDY interrupt mode.	Pipes 1 to 5: Settable
	DBLB	Selects double-buffer configuration.	Pipes 1 to 5: Settable
	CNTMD	Selects continuous transfer or non-continuous transfer.	DCP: Settable Pipes 1, 2: Settable only in bulk transfers Pipes 3 to 5: Settable
	DIR	Selects transfer direction.	Settable to IN or OUT
	EPNUM	Endpoint number	Pipes 1 to 9: Settable Set to a value other than “0000” when a pipe is in use.
	SHTNAK	Disables pipes when transfer is completed.	DCP: Settable Pipes 1, 2: Settable only in bulk transfers Pipes 3 to 5 Settable
PIPEBUF	BUFSIZE	Buffer memory size	DCP: Unsettable (fixed to 64/256 (CNTMD = 1) bytes) Pipes 1 to 5: Settable (up to 2 Kbytes specifiable) Pipes 6 to 9: Unsettable (fixed to 64 bytes)
	BUFNMB	Buffer memory number	DCP: Unsettable (fixed to area 0 to 3 hex) Pipes 1 to 5 Settable (area 8 to 87 hex specifiable) Pipes 6 to 9: Unsettable (fixed to area 4 to 7 hex)
DCPMAXP PIPEMAXP	MXPS	Maximum packet size	Setting conforming to the USB specification.
PIPEPERI	IFIS	Buffer flush	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 5: Unsettable Pipes 6 to 9: Unsettable
	IITV	Interval counter	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 5: Unsettable Pipes 6 to 9: Unsettable
DCPCTR PIPExCTR	BSTS	Buffer status	DCP state switched between receive and transmit buffer by ISEL bit
	INBUFM	IN buffer monitor	Available only for pipes 3 to 5
	ACLRM	Auto buffer clear	Pipes 1 to 9: Settable
	SQCLR	Sequence clear	Clears data toggle bit.
	SQSET	Sequence set	Sets data toggle bit.
	SQMON	Sequence monitor	Monitors data toggle bit.
	PBUSY	Pipe busy monitor	
	PID	Response PID	
DCPCTR PIPExCTR	ATREPM	Auto response mode	Pipes 1 to 5: Settable
PIPExTRE	TRENB	Transaction count enable	Pipes 1 to 5: Settable
	TRCLR	Current transaction counter clear	Pipes 1 to 5: Settable
PIPExTRN	TRNCNT	Transaction counter	Pipes 1 to 5: Settable

32B.9.3.1 Maximum packet size setting

Use the MXPS bits in the DCPMAXP and PIPEMAXP registers to specify the maximum packet size for each pipe. The default control pipe (DCP) and pipes 1 to 5 can be set to any of the maximum packet sizes defined by the USB specification. For pipes 6 to 9, 64 bytes are the upper limit of the maximum packet size. Set the maximum packet size before starting transfer (by setting “PID = BUF”).

DCP: Set 64 for high-speed operation.

DCP: Set 8, 16, 32, or 64 for full-speed operation. Pipes 1 to 5: Set 512 for high-speed bulk transfer.

Pipes 1 to 5: Set 8, 16, 32, or 64 for full-speed bulk transfer.

Pipes 1, 2: Set a value from 1 to 1024 for high-speed isochronous transfer.

Pipes 1, 2: Set a value from 1 to 1023 for full-speed isochronous transfer.

Pipes 6 to 9: Set 64.

High-bandwidth transfers used for interrupt transfers and isochronous transfers are not supported.

32B.9.3.2 Response PID

Set the response PID for each pipe with the PID bits in the DCPCTR and PIPExCTR registers.

(1) Response PID setting
The response PID specifies the response to a transaction from the host.

a) NAK: Always sends a NAK response to a generated transaction.

b) BUF: Responds to a transaction in accordance with the buffer memory state.

c) STALL: Always sends a STALL response to a generated transaction.

Regardless of the value set in the PID bits, an ACK is always sent as a response to a setup transaction and a USB request is stored in corresponding registers.

This module might write data to the PID bits depending on the transaction result. This module writes data to the PID bits in the following cases:

a) NAK:

(A) The SETUP token is received normally (for the DCP only).

(B) If 1 is set for the SHTNAK bit in the PIPECFG register during bulk transfer, a short packet is received.

(C) If 1 is set for the SHTNAK bit during bulk transfer, the transaction counter finishes.

b) BUF:

This module does not write "BUF".

c) STALL:

(A) When a maximum packet size over error is detected in a received data packet

(B) When a control transfer sequence error is detected

32B.9.3.3 Pipe control register switching procedure

The following bits in the pipe control registers can be modified only when USB transmission is disabled (PID = NAK).

Figure 32B.8 shows the procedure for changing the pipe control register state from the USB transmission enabled (PID = BUF) state.

The registered that cannot be manipulated in the USB transmission enabled (PID = BUF) state are as follows:

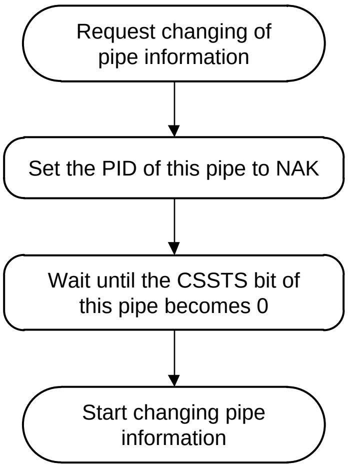
(1) All bits in the DCPCFG and DCPMAXP registers

(2) SQCLR and SQSET bits in the DCPCTR register

(3) All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers

(4) ATREPM, ACLRM, SQCLR, and SQSET bits in the PIPExCTR register

(5) All bits in the PIPExTRE and PIPExTRN registers



Note: If the pipe is detached during a USB transaction, the PBUSY bit might remain 1.

Figure 32B.8 Procedure for Changing Pipe Information from USB Transmission Enabled (PID = BUF) State

In addition, for the settings of the following bits in pipe control registers, only the pipe information that is not set for the CURPIPE bit in the CPU, DMA0, or DMA1 FIFO port can be changed.

Registers that cannot be set with pipe information that is set for the CURPIPE bit in a FIFO port:

(6) All bits in the DCPCFG and DCPMAXP registers

(7) All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers

(8) ACLRM bit in the PIPExCTR register

If you change the pipe information, make sure that the setting of CURPIPE is different from the new pipe number. For the default control pipe (DCP), after modifying the pipe information, use the BCLR bit to clear the buffer.

32B.9.3.4 Data PID sequence bit

When a normal data transfer is performed during bulk transfer or interrupt transfer, or in the data stage of control transfer, this module automatically toggles the sequence bit of a data PID. The sequence bit of the next data PID to be transferred can be confirmed with the SQMON bit in the DCPCTR or PIPExCTR registers. For data transmission, the sequence bit is switched when an ACK handshake is received. For data reception, the sequence bit is switched when an ACK handshake is sent. The data PID sequence bit can be changed by using the SQCLR and SQSET bits in the DCPCTR and PIPExCTR registers.

During control transfer, this module automatically sets the sequence bit when the stage changes. When the setup stage finishes, the data PID is set to DATA1. In the status stage, this module responds with “PID = DATA1” without referencing the sequence bit.

Note that, the data PID sequence bit must set, when a clear feature request is sent or received.

Also note that for isochronous transfer setting pipes, you cannot use the SQSET bit to manipulate the sequence bit.

32B.9.4 FIFO Buffer

This section describes the processing related to the FIFO buffer of this module.

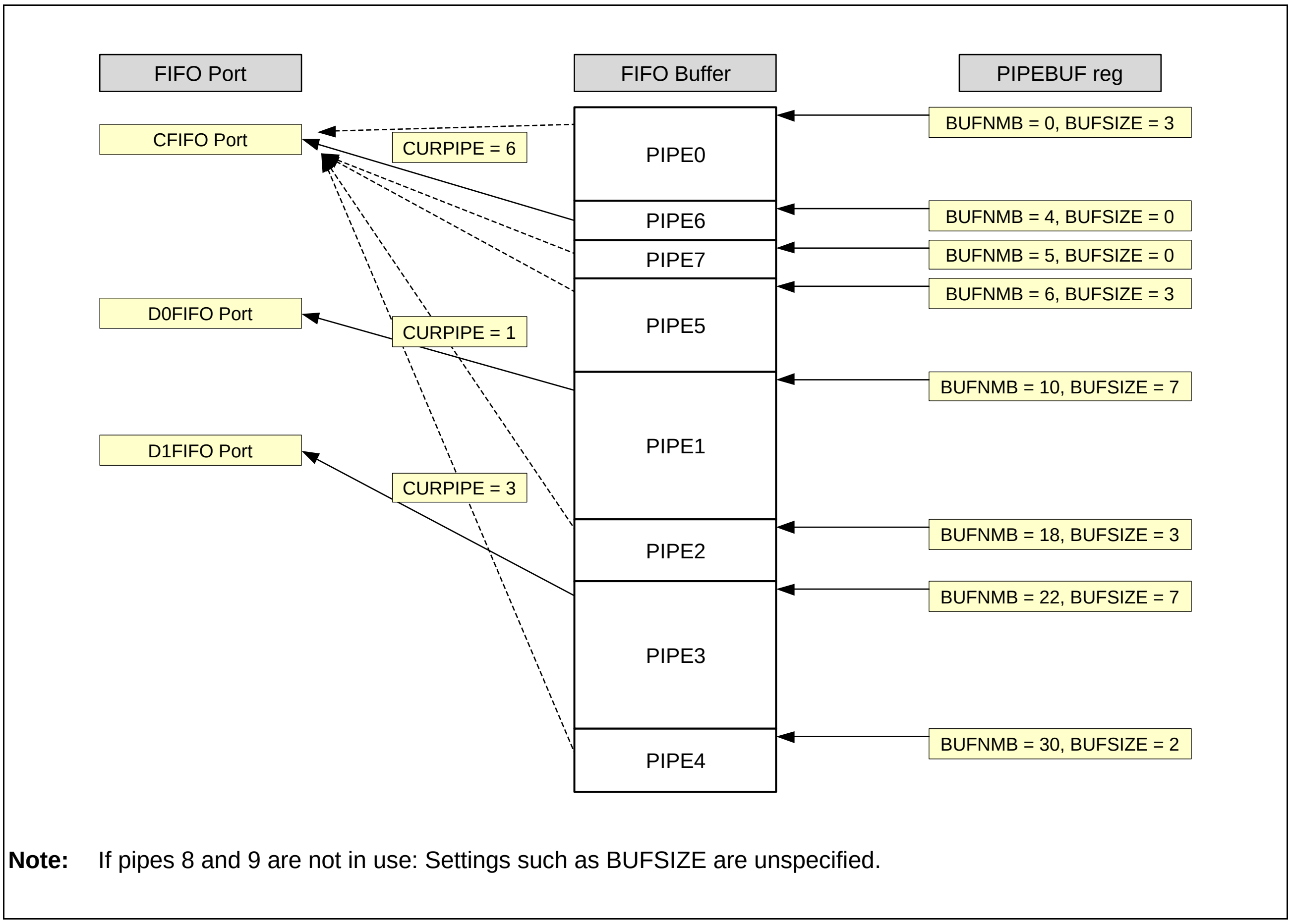
32B.9.4.1 FIFO buffer allocation

Figure 32B.9 shows an example of FIFO buffer memory mapping of this module. The FIFO buffer area is shared by the CPU and this module. The FIFO buffers can be accessed by either the system (CPU) or this module (SIE).

An independent FIFO buffer area is allocated for each pipe. The memory area is made up of memory blocks (1 block = 64 bytes) and is specified by the starting block number and the number of blocks (specified by the BUFNMB and BUFSIZE bits in the PIPEBUF register). When the CNTMD bit in the PIPExCFG register is used to set "continuous transfer mode", the value set with the BUFSIZE bit must be an integral multiple of the maximum packet size. If the double-buffer configuration is selected by the DBLB bit in the PIPExCFG register, two memory areas the size of which is specified by the BUFSIZE bit in the PIPEBUF register are allocated to a single pipe.

FIFO ports are used to access the FIFO buffer (data read/write). The pipe number of a pipe to be assigned to a FIFO port is specified by the CURPIPE bits in the CFIFOSEL/DxFIFOSEL register.

The FIFO buffer status of each pipe can be confirmed by using the BSTS and INBUFM bits in the DCPCTR and PIPExCTR registers. The access right of a FIFO port can be confirmed by using the FRDY bit in the CFIFOCTR/DxFIFOCTR register.



Note: If pipes 8 and 9 are not in use: Settings such as BUFSIZE are unspecified.

Figure 32B.9 Example of FIFO Buffer Memory Mapping

32B.9.4.2 Clearing FIFO buffers

Table 32B.25 lists the modes in which this module can clear the FIFO buffer. Clearing of the FIFO buffer is controlled by the following bits.

Table 32B.25 List of FIFO Buffer Clearing Modes

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR register DxFIFOCTR register	DxFIFOSEL register	PIPExCTR register
Function	The FIFO buffer on the CPU side is cleared.	The FIFO buffer is automatically cleared after the data is read from the specified pipe.	The buffer is automatically cleared to discard all the received packets.
Clearing method	Write “1” to clear.	1: Mode enabled 0: Mode disabled	1: Mode enabled 0: Mode disabled

32B.9.5 FIFO Port Functions

This section describes FIFO port functions. **Table 32B.26** shows the FIFO port function settings for this module.

If data write access is enabled and data is written up to buffer full state (in non-continuous transfer: maximum packet size), the port automatically goes to the USB bus transmittable state. To enable transmission of data smaller than the buffer full (in non-continuous transfer: less than the maximum packet size), the BVAL bit in the CFIFOCTR/ DxFIFOCTR register must be used to set write end.

To send a zero-length packet, the BCLR bit of that register must be used to clear the buffer before the BVAL bit is used to write end.

When all the data is read in a read access, the port automatically enters the state in which new packets can be received. However, when a zero-length packet is received (DTLN = 0), no data can be read, and therefore the buffer must be cleared by using the BCLR bit.

The receive data length is confirmed with the DTLN bits in the CFIFOCTR/DxFIFOCTR register.

Table 32B.26 FIFO Port Function Settings

Register Name	Bit Name	Function	Note
C/DxFIFOSEL	RCNT	Selects DTLN read mode	
	REW	Rewinds buffer memory (re-read, re-write)	
	DCLRM	Reads received data of the specified pipe, and then automatically clears the received data	DxFIFO only
	DREQE	Asserts DREQ signal	DxFIFO only
	MBW	Specifies FIFO port access bit width	
	BIGEND	Selects FIFO port endian	CFIFO only
	ISEL	Specifies FIFO port access direction	DCP only
	CURPIPE	Selects current pipe	
C/DxFIFOCTR	BVAL	Finishes buffer memory write	
	BCLR	Clears CPU-side buffer memory	
	FRDY	Monitors FIFO port ready	
	DTLN	Confirms received data length	

32B.9.5.1 FIFO port selection

Table 32B.27 lists pipes that can be selected for each FIFO port.

Use the CURPIPE bits in the C/DxFIFOSEL register to select the pipe to be accessed. After selecting the pipe, confirm that the value written to the CURPIPE bits can be read correctly (if the previous pipe number is read out, this module is currently changing the pipe), confirm that FRDY = 1, and then access the FIFO port.

Figure 32B.10 shows the procedure for switching the pipe when accessing a FIFO port.

In addition, use the MBW bit to select the bus width with which to access the FIFO port. If the target pipe is the default control pile (DCP), the ISEL bit determines the buffer memory access direction. If the target pipe is not the DCP, the DIR bit in the PIPExCFG register determines the buffer memory access direction.

Table 32B.27 FIFO Port Access for Each Pipe

Pipe	Access Method	Usable Port
DCP	CPU access	CFIFO port register
Pipes 1 to 9	CPU access	CFIFO port register
	DMA access	DxFIFO port register

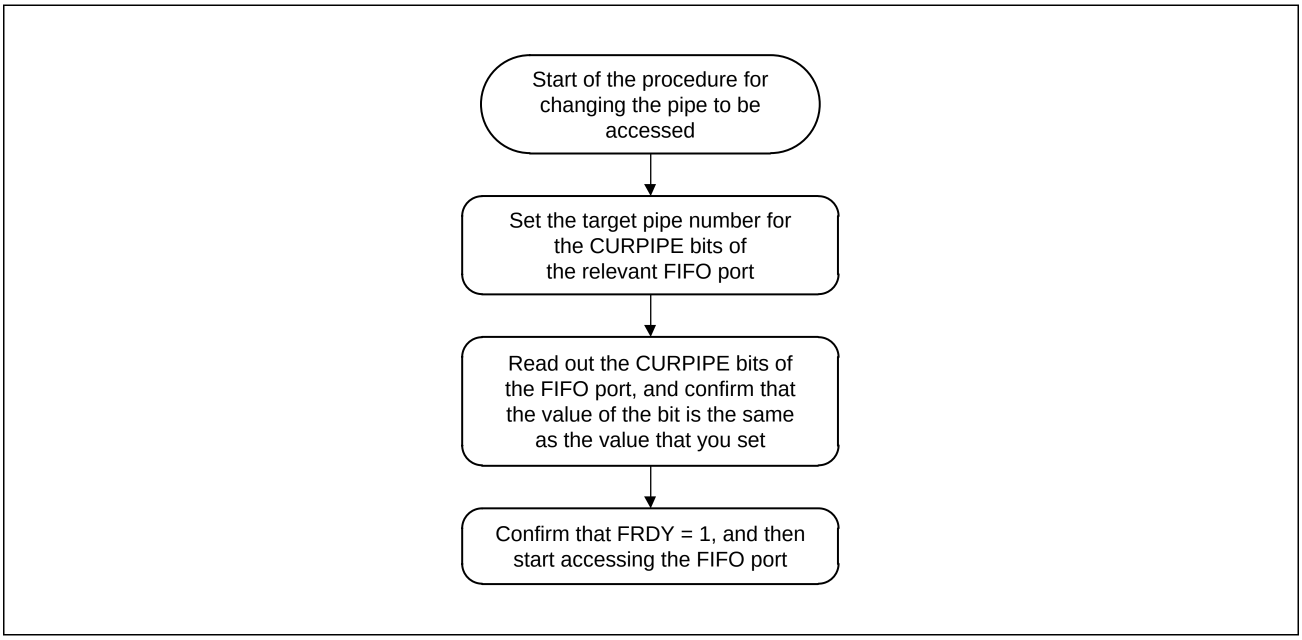


Figure 32B.10 Pipe Switching Procedure for FIFO Port Access

32B.9.5.2 DxFIFO automatic clear mode (DxFIFO port read direction)

This module automatically clears the buffer memory for a pipe when data is read out from the buffer memory if the DCLRM bit in the DxFIFOSEL register is set to 1.

Table 32B.28 shows the correspondence between packet reception and buffer memory clear processing by the software for each setting.

As shown in **Table 32B.28**, the buffer clearing conditions vary with the value that is set for the BFRE bit. However, using the DCLRM bit eliminates the need for buffer clear by the software even in states where clearing is required, which enables DMA transfers without using the software.

Note that for this function, only the buffer memory read direction can be set.

Table 32B.28 Relationship between Packet Reception and Buffer Memory Clear Processing by the Software

	Register Setting			
	DCLRM = 0		DCLRM = 1	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Clearing unnecessary		Clearing unnecessary	
Zero-length packet received	Clearing necessary			
Normal short packet received	Clearing unnecessary	Clearing necessary		
Transaction count end				

32B.9.5.3 BRDY interrupt timing selection function

The BFRE bit in the PIPECFG register can be set so that a BRDY interrupt is not generated when a data packet of maximum packet size is received.

When using a DMA transfer, this function enables an interrupt to be generated only when the last data is received. The “last data” here indicates either a short packet reception or the transaction count end. If the BFRE bit is set to 1, a BRDY interrupt is generated after the received data is read. By reading the DTLN bits in the DxFIFOCTR register, the receive data length of last data packet received just before the BRDY interrupt is generated can be confirmed.

Table 32B.29 shows when this module generates a BRDY interrupt.

Table 32B.29 BRDY Interrupt Generation Timing

	Register Setting	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1
Buffer full (normal packet received)	When packet is received	No interrupt generated
Zero-length packet received	When packet is received	When packet is received
Normal short packet received	When packet is received	When received data has been read from the buffer memory
Transaction count end	When packet is received	When received data has been read from the buffer memory

The BFRE bit function is valid only in the reading direction from the buffer memory. For the writing direction, fix the BFRE bit to 0.

32B.9.6 Control Transfer (DCP)

In the data stage of control transfer, the default control pipe (DCP) is used to transfer data. For the DCP, a single 64-byte buffer is allocated as a fixed area that is used for both control reading and writing (in continuous transfer mode (CNTMD = 1), the size of this area is fixed to 256 bytes). The buffer memory can be accessed through the CFIFO port only.

32B.9.6.1 Control transfer

(1) Setup stage

This module always responds with ACK to any normal setup packet for this module. The following shows the behavior of this module in the setup stage:

1. When this module receives a new setup packet, this module sets the following bits:
    (a) VALID bit in the INTSTS0 register: 1
    (b) PID bit in the DCPCTR register: NAK
    (c) CCPL bit in the DCPCTR register: 0

2. When this module receives a data packet after receiving a setup packet, this module stores USB request parameters in the USBREQ, USBVAL, USBINDX, and USBLENG registers.

Response processing for control transfer must be performed after VALID is set to 0. While VALID is 1, PID cannot be set to BUF, and therefore the data stage cannot end.

By using a function of the VALID bit, when this module receives a new USB request during control transfer, this module can respond to the newest request, canceling the request that is being processed.

This module also automatically recognizes the type of transfer (ControlRead, ControlWrite, or NodataControl) from the direction bit (bit 8 of bmRequestType) and request data length (wLength) of the received USB request to manage stage transition. For an incorrect sequence, a control transfer stage transition interrupt occurs to report a sequence error to software. For details about stage management by this module, see **Figure 32B.7**.

(2) Data stage

Use the DCP to transfer data in response to the received USB request. Before the DCP buffer memory is accessed, use the ISEL bit in the CFIFOSEL register to specify the access direction.

A transaction is executed by setting the PID bit in the DCPCTR register to BUF.

The end of data transfer is detected with a BRDY or BEMP interrupt. For ControlWrite transfer, use a BRDY interrupt. For ControlRead transfer, use a BEMP interrupt.

For ControlWrite transfer in high-speed operation mode, an NYET handshake response is performed in accordance with the buffer memory state.

(3) Status stage

If the PID bit in the DCPCTR register is BUF, setting the CCPL bit to 1 terminates control transfer.

After control transfer is terminated by the above setting, this module automatically executes the status stage according to the data transfer direction determined in the setup stage, as shown below:

(a) For ControlRead transfer

Upon receiving a zero-length packet from the USB Host Controller, this module sends an ACK response.

(b) For ControlWrite or NodataControl transfer

This module sends a zero-length packet, and then receives an ACK response from the USB Host Controller.

(4) Control transfer automatic response function

This module automatically responds to any normal SET_ADDRESS request. However, if a SET_ADDRESS request has any of the following errors, software must respond, instead of this module:

(a) bmRequestType ≠ "H'00"

(b) wIndex ≠ "H'00"

(c) wLength ≠ "H'00"

(d) wValue > "H'7F"

(e) DVSQ = "011 (Configured) "

Software must respond to all requests other than SET_ADDRESS.

32B.9.7 Bulk Transfer (Pipes 1 to 5)

The user can select the buffer memory usage method (single/double buffer, continuous/non-continuous transfer mode) for bulk transfers. The buffer memory size can be set up to 2 KB. The controller manages the buffer memory state and automatically responds to PING packets and NYET handshakes.

32B.9.7.1 NYET handshake control

Table 32B.30 lists responses to tokens received in a bulk or control transfer.

When an OUT token is received in a bulk or control transfer and there is only an open space for one packet in the buffer memory, this module sends a NYET response. However, when a short packet is received, this module sends an ACK response instead of a NYET response even under these conditions.

Table 32B.30 List of Responses to Received Tokens

PID bit value	Buffer memory state*[1]	Received token	Response	Note
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY	OUT/PING	ACK	Receives data packet when OUT token is received*[1]
	RCV-BRDY	OUT	NYET	Receives data packet*[2]
	RCV-BRDY	OUT (Short)	ACK	Receives data packet*[2]
	RCV-BRDY	PING	ACK	*[2]
	RCV-NRDY	OUT/PING	NAK	
	TRN-BRDY	IN	DATA0/1	Sends data packet
	TRN-NRDY	IN	NAK	

Note: Details are described below.
RCV-BRDY*1: Buffer memory has a space for 2 packets or more when an OUT or PING token is received.
RCV-BRDY*2: Buffer memory has only a space for one packet when an OUT token is received.
RCV-NRDY: Buffer memory has no space for any packet when a PING token is received.
TRN-BRDY: Buffer memory has sent data when an IN token is received.
TRN-NRDY: Buffer memory has no send data when an IN token is received.

32B.9.8 Interrupt Transfer (Pipes 6 to 9)

This module performs an interrupt transfer in accordance with the period managed by the host controller. This module ignores (no response) PING packets in interrupt transfers. In addition, this module does not send a NYET handshake, but sends an ACK, NAK or STALL response.

Note that this module does not support high-bandwidth interrupt transfers.

32B.9.9 Isochronous Transfer (Pipes 1 and 2)

This module is provided with the following functions for isochronous transfers:

1. Notification of error information about isochronous transfers
2. Interval counter (IITV bit)
3. Data setup control for isochronous IN transfers (IDLY function)
4. Buffer flush function for isochronous IN transfers (IFIS bit)
5. SOF pulse output function

This module does not support high-bandwidth Isochronous transfers.

32B.9.9.1 Isochronous transfer error detection

This module has the following error information detection functions for the software to manage errors that occur during isochronous transfer.

Table 32B.31 and **Table 32B.32** describe the error checking procedure and interrupts that are generated.

1. PID error
   The PID of the received packet is invalid.
2. CRC error and bit stuffing error
   The received packet has a CRC error or invalid bit stuffing.
3. Max packet size over
   The data size of the received packet is larger than the preset maximum packet size.
4. Overrun error and underrun error
   (a) The buffer memory has no data when an IN token is received during IN-direction (send) transfer.
   (b) The buffer memory has no space although an OUT token is received during OUT-direction (receive) transfer.
5. Interval error
   An interval error occurs in the following cases:
   (a) An IN token cannot be received within the interval frame during isochronous IN transfer.
   (b) An OUT token cannot be received within the interval frame during isochronous OUT transfer.

Table 32B.31 Error Detection during Transmission/Reception of Token

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	No interrupt is generated (ignored as a corrupted packet).
3	Overrun error, underrun error	An NRDY interrupt is generated and the OVRN bit is set. A zero-length packet is sent in response to an IN token. A data packet is not received in response to an OUT token.
4	Interval error	An NRDY interrupt is not generated.

Table 32B.32 Error Detection during Reception of Data Packet

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	An NRDY interrupt is generated and the CRCE bit is set.
3	Packet size error (too large packet)	A BEMP interrupt is generated and the PID bit is set to STALL.

32B.9.9.2 DATA-PID

This module does not support high-bandwidth transfers.

The following shows actions that can be taken in response to a received PID.

1. IN direction:

(a) DATA0: Used to send packets.

(b) DATA1: Not used to send packets.

(c) DATA2: Not used to send packets.

(d) mDATA: Not used to send packets.

2. OUT direction (in full-speed operation):

(a) DATA0: Packets are received normally.

(b) DATA1: Packets are received normally.

(c) DATA2: Packets are ignored.

(d) mDATA: Packets are ignored.

3. OUT direction (in high-speed operation):

(a) DATA0: Packets are received normally.

(b) DATA1: Packets are received normally.

(c) DATA2: Packets are received normally.

(d) mDATA: Packets are received normally.

32B.9.9.3 Interval counter

(1) Outline of operation

The IITV bit in the PIPEPERI register can be used to set the interval of isochronous transfer. The interval counter enables the functions listed in **Table 32B.33**.

Table 32B.33 Functions of the Interval Counter

Transfer Direction	Function	Detecting Condition
IN	Transmit buffer flush function	An IN token cannot successfully be received within the interval frame during isochronous IN transfer.
OUT	Notification of unreceived token	An OUT token cannot successfully be received within the interval frame during isochronous OUT transfer.

Counting of intervals is based on received SOF packets or interpolated SOFs. Therefore, even if SOF packets are damaged, the isochronism can still be maintained. Frame intervals are set as 2n (micro) frames, where n is the value of the IITV bit.

(2) Interval counter initialization

This module initializes the interval counter under the following conditions:

(a) Power on reset
The IITV bit is initialized.

(b) Clearing of the buffer memory by the ACLRM bit
The IITV bit is not initialized but the counter is initialized.

(c) USB bus reset

After the interval counter is initialized and a packet is successfully transferred, counting of intervals starts under the following conditions:

1) An SOF packet is received after data is sent in response to an IN token when PID = BUF.

2) An SOF packet is received after data is received in response to an OUT token when PID = BUF.

Note that the interval counter is not initialized in the following conditions:

(a) The PID bit is set to NAK or STALL.
The interval timer is not stopped at this interval. The transaction will be attempted at the next interval.

(b) USB bus reset or USB suspension
The IITV bit is not initialized. When an SOF packet is received, counting starts from the value existing before reception.

32B.9.9.4 Send data setup for isochronous transfer

This module becomes able to send data packets by isochronous transfer from the next frame after data is written to the buffer memory and then an SOF packet is detected. This is called “send data setup for isochronous transfer”.

This function can identify the frame with which data sending started.

If the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, only the buffer to which writing finished earlier can transfer data. Therefore, even when several IN tokens are received within the same frame, only one packet of data is sent from the buffer memory.

When an IN token is received, if the buffer memory is ready for sending data, the data is transferred and a normal response is returned. However, if the buffer memory is not ready for sending data, a zero-length packet is sent and an underrun error occurs.

Figure 32B.11 shows examples of sending using the send data setup function for isochronous transfer by setting “IITV = 0” (each frame) in this module.

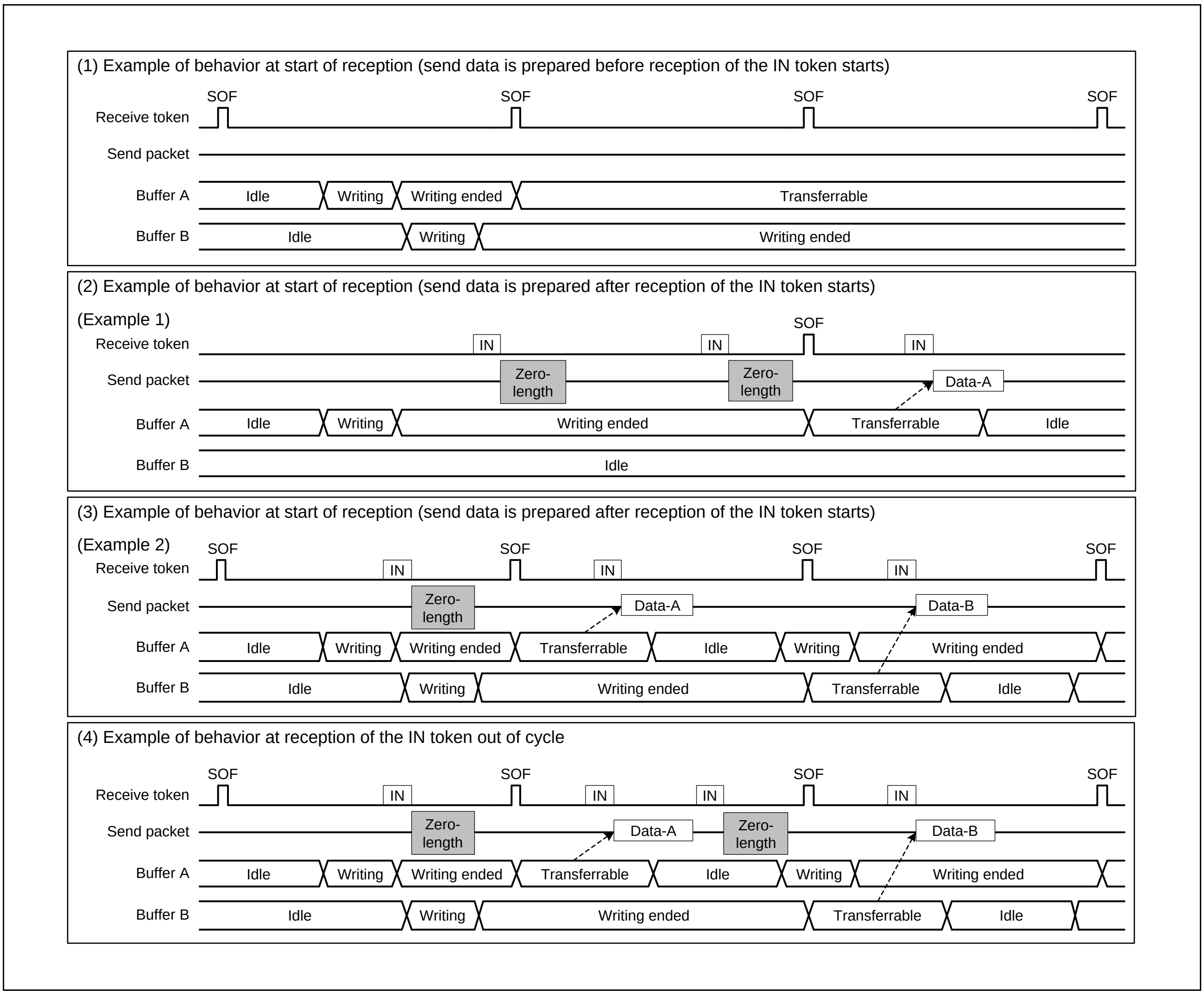


Figure 32B.11 Examples of Data Setup Function Behavior

32B.9.9.5 Transmit buffer flush for isochronous transfer

If this module does not receive an IN token in an interval frame and receives a (micro) SOF packet in the next frame during isochronous data transfer, this module handles the IN token as a corrupted token and clears the buffer that can send data to make the buffer writable.

At this time, if the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, this module assumes the discarded buffer memory to be sent within the same interval frame. As a result, the buffer memory that is not discarded by reception of a (micro) SOF packet becomes to be able to transfer data.

The operation start timing of the buffer flush function varies with the value of the IITV bit.

1. If IITV is 0
   Buffer flush operation is performed from the first frame after the pipe is enabled.
2. If IITV is not 0
   Buffer flush operation is performed after the first successful transaction.

Figure 32B.12 shows an example of how the buffer flush function of this module behaves. For a token outside the set interval (token prior to the interval frame), however, this module sends the written data or a zero-length packet as an underrun error according to the data setup state.

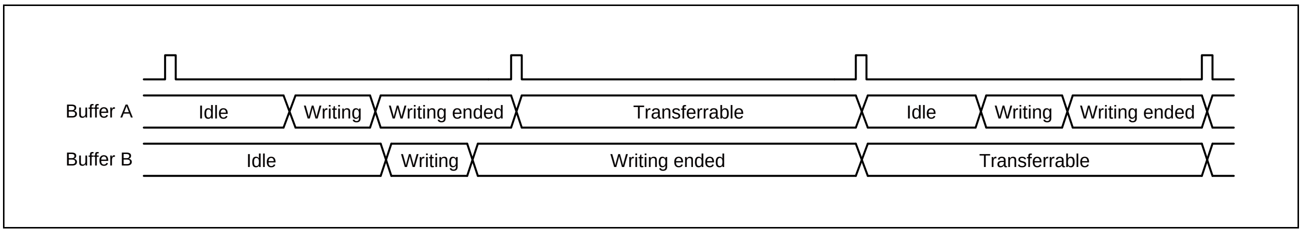


Figure 32B.12 Example of Buffer Flush Function Behavior

Figure 32B.13 shows an example of an interval error that occurs in this module.

There are five types of interval errors, as listed below. At timing (1) in the figure, an interval error occurs and the buffer flush function operates.

If an interval error occurs during IN transfer, the buffer flush function starts. If an interval error occurs during OUT transfer, an NRDY interrupt occurs.

Use the OVRN bit to determine whether an error is an NRDY interrupt (such as a receive packet error) or an overrun error.

In the figure, responses to tokens indicated as shaded boxes are made in accordance with the buffer memory state.

1. IN direction:

(a) If the buffer is ready to transfer data, data is transferred and a normal response is returned.

(b) If the buffer is not ready to transfer data, a zero-length packet is sent and an underrun error occurs.

2. OUT direction:

(a) If the buffer is ready to receive data, data is received and a normal response is returned.

(b) If the buffer is not ready to receive data, data is discarded and an overrun error occurs.

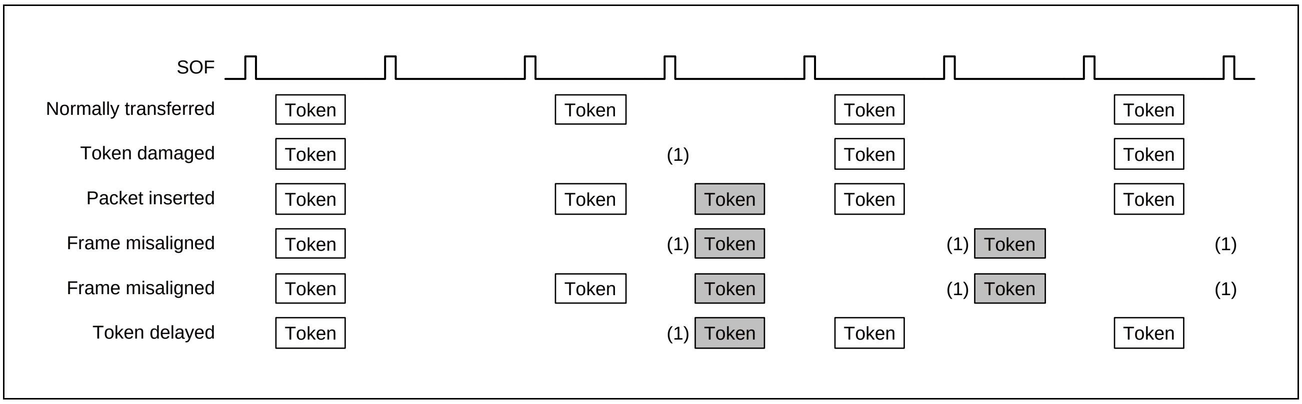


Figure 32B.13 Example of Occurrence of Interval Error (when IITV is 1)

32B.9.10 SOF Interpolation Function

If the controller cannot receive data at intervals of 1 ms (in full-speed operation) or 125 µs (in high-speed operation) due to corruption or missing of an SOF packet, the controller internally interpolates the SOF. The controller starts SOF interpolation upon receiving an SOF packet when both the USBE bit and SUSPM bit are set to 1.

The interpolation function is initialized under the following conditions:

(1) Power on reset

(2) USB bus reset

(3) Suspended state detected

The SOF interpolation operates according to the following specifications:

(1) Frame interval (125 µs or 1 ms) is based on the results of the reset handshake protocol.

(2) The interpolation function does not operate until an SOF packet is received.

(3) After receiving the first SOF packet, this module interpolates the SOF by using the 60-MHz internal clock to measure 125 µs or 1 ms.

(4) After receiving the second or a subsequent SOF packet, this module interpolates the SOF by using the previous reception interval.

(5) Interpolation is not performed in the suspended state or while a USB bus reset is being received.
(If this module enters the suspended state in high-speed operation, interpolation continues for 3 ms after receiving the last packet.)

The SOF interpolation function operates with the following functions:

(1) Updating of frame number or micro-frame number

(2) SOFR interrupt and micro SOF lock

(3) SOF pulse output

(4) Counting of isochronous transfer intervals

If an SOF packet is lost in full-speed operation, the FRNM bit in the FRMNUM register is not updated.

If a micro SOF packet is lost in high-speed operation, the UFRNM bit in the uFRMNUM register is updated.

However, if a micro SOF packet for which “micro-FRNM = 000” is set is lost, the FRNM bit is not updated. In this case, the FRNM bit is not updated even if subsequent micro SOF packets for which “micro-FRNM = 000” is not set are successfully received.

32B.9.11 Link Power Management Processing

According to the Link Power Management specification, the existing suspend state is redefined as the L2 state and a new L1 state is defined as a state where transition and resumption at a lower latency than L2 (suspend) are possible.

The table below compares the features of the L2 (suspend) state and L1 state.

Table 32B.34 Comparison Between Suspend (L2) State and L1 State

Item	L1	Suspend (L2)
Transition	LPM Transaction	3-ms idle period
Host-activated resumption	(Host) Minimum drive period specifiable by the host. Specified between 75 µs to 1.175 ms (Device) 10-µs K drive	(Host) Min. 20-ms K drive (Device) 10-ms K drive
Device-activated resumption	(Device) 50-µs K drive (Host) 60- to 990-µs K drive (Device) 10-µs K drive	(Device) 1-ms to 15-ms K drive (Host) Min. 20-ms K drive (Device) 10-ms K drive
Signaling	Low and Full Speed Idle	Low and Full Speed Idle

The following describes the processing for transition to and resumption from the L1 state.

32B.9.11.1 Descriptor

This module must return its own descriptor when receiving the GetDescriptor command.

Whether the contents of the descriptor to be returned need to be modified is dependent on whether this module responds to the transition to and resumption from the L1 state with an LPM transaction. The details are summarized in the table below.

Table 32B.35 Relationship between LPM Response and Descriptor

LPM Response	bcdUSB	Presence of USB 2.0 Ex.Desc	USB 2.0 Ex.Desc LPM	Response When LPM Is Received	Remarks
Not respond	0200	Not present	—	Not Respond	Standard action in the case where this module does not respond to LPM
	0201	Present	LPM = 0	STALL	This is when rejection of response to LPM is explicitly declared. In this case, it is necessary to send a STALL response instead of making no response.
Respond	0201	Present	LPM = 1	ACK or NYET	Standard action in the case where this module responds to LPM

Whether to respond to transition to and resumption from L1 is declared by the LPM bit of the USB 2.0 extension descriptor. To provide this module with the USB 2.0 extension descriptor, it is necessary to set the bcdUSB field of the device descriptor to 0201 or greater.

When not responding to LPM, set the bcdUSB value to 0200 without providing this module with the USB 2.0 extension descriptor. In this case, it is necessary to ignore any LPM token received.

When not responding to LPM, it is also possible to set bcdUSB to 0201 and set the LPM bit of the USB 2.0 extension descriptor to 0 (noncompliant). In this case, however, it is not allowed to ignore LPM and is necessary to send a STALL response.

When responding to LPM, set bcdUSB to 0201 and the LPM bit of the USB 2.0 extension descriptor to 1 (compliant). This grants this module to send an NYET or ACK in response to an LPM token.

32B.9.11.2 Basic processing

This module needs to execute the following processing.

1. Responds to the LPM token received from the host with “No response”, “ACK”, “NYET”, or “STALL” according to this module's own state.
2. Transitions to the L1 state if it fails to detect the retransmission of an LPM token for 8 µs after making an ACK response.
3. Detects a K drive of the host and performs resume processing to the idle state.
4. Performs resume processing to the idle state based on the Remote Wake signal.

For 1, the software specifies the response method according to the values of the L1RESPEN, L1RESPMD, and L1NEGOMD bits in the PL1CTRL register. The hardware makes the response that is designated by the software upon receiving an LPM token.

For 2, both retransmission control and transition to the L1 state are processed by the hardware. Transition to the L1 state can be identified through a DVST interrupt.

For 3, a RESM interrupt occurs on detection of host K in the L1 state.

For 4, starting the Remote Wake processing can be instructed to the hardware by setting the WKUP bit by the software. The specification stipulates that the software clears this bit on resumption from the L2 state. On the other hand, the hardware clears this bit on resumption from the L1 state.

32B.9.11.3 HIRD value negotiation

The HIRD value contained in the LPM token is the K period of the host on resumption from the L1 state.

This module can respond with ACK if the received HIRD value falls within the desired range as specified by the L1NEGOMD and HIRDTHR bits in the L1CTRL register; otherwise, this module can respond with NYET and request the host to modify the HIRD value.

NOTE

This HIRD value negotiation function must also be supported on the host side.

32B.9.12 DMA Mode

32B.9.12.1 Register mode/link mode

The DMS bit in the CHCFG_n register can be used to switch between register mode and link mode.

Table 32B.36 DMA mode settings

DMS (CHCFG)	Mode	Description
0	Register mode	Performs DMA transfer based on the values set by Next Register Set.
1	Link mode	Accesses the descriptor area, and executes DMA transfer based on the values set by descriptors. This module repeats descriptor reading and DMA transfer unless the descriptor settings are changed or the control register is used to stop the processing.

(1) Register mode

In register mode, this module executes DMA transfer based on the values set in internal registers.

Two sets of the transfer-source address, transfer-destination address, and number of bytes to be transferred can be held (in Next0 Register Set and Next1 Register Set registers). One of these Next registers can be used to execute transfer, and both Next registers can be used to execute continuous transfer.

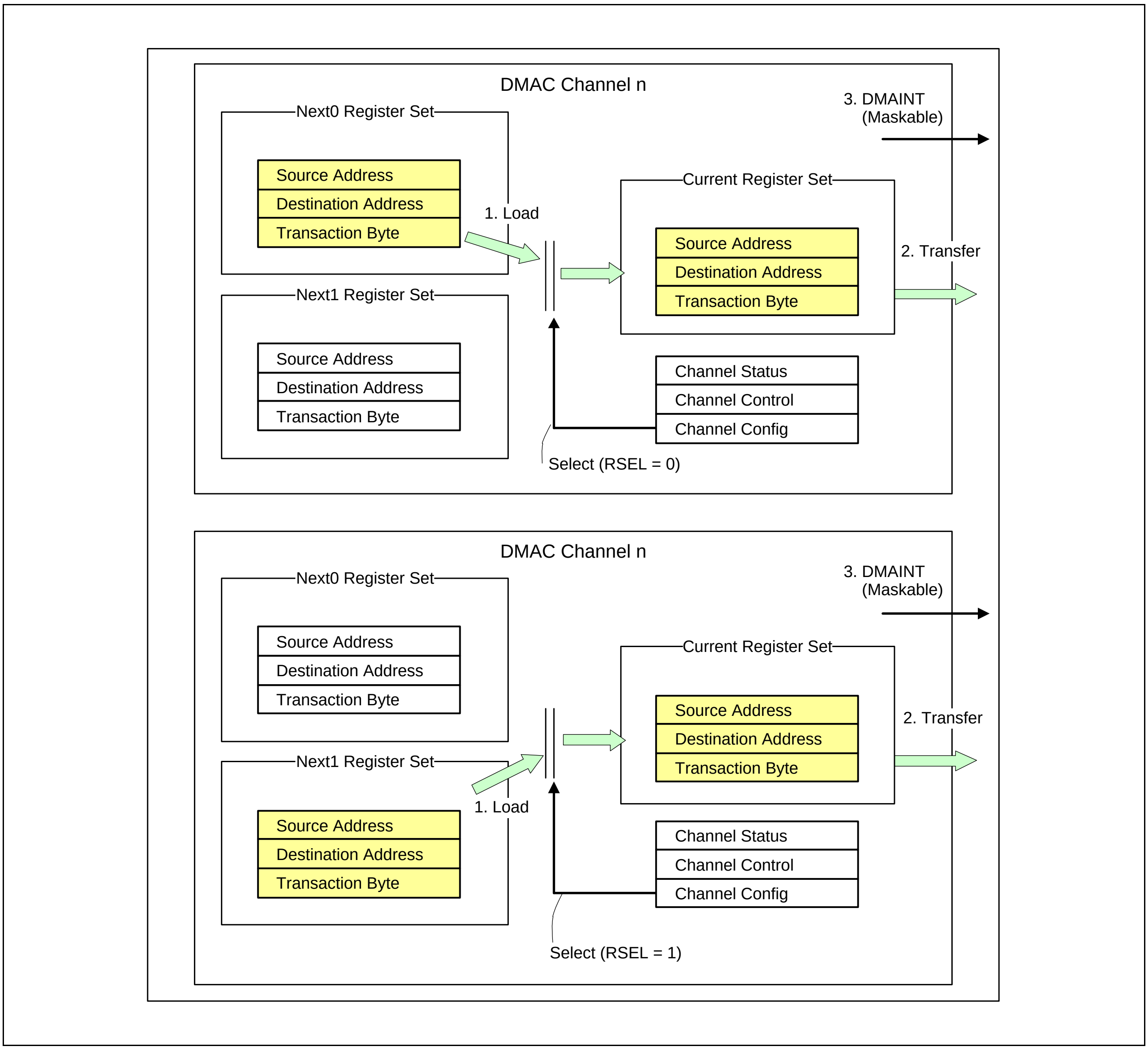


Figure 32B.14 Overview of Normal Behavior of Register Mode

The upper part of the above figure indicates a case when Next0 Register Set is processed. The lower part of the above figure indicates a case when Next1 Register Set is processed.

(a) Operation flow of register mode

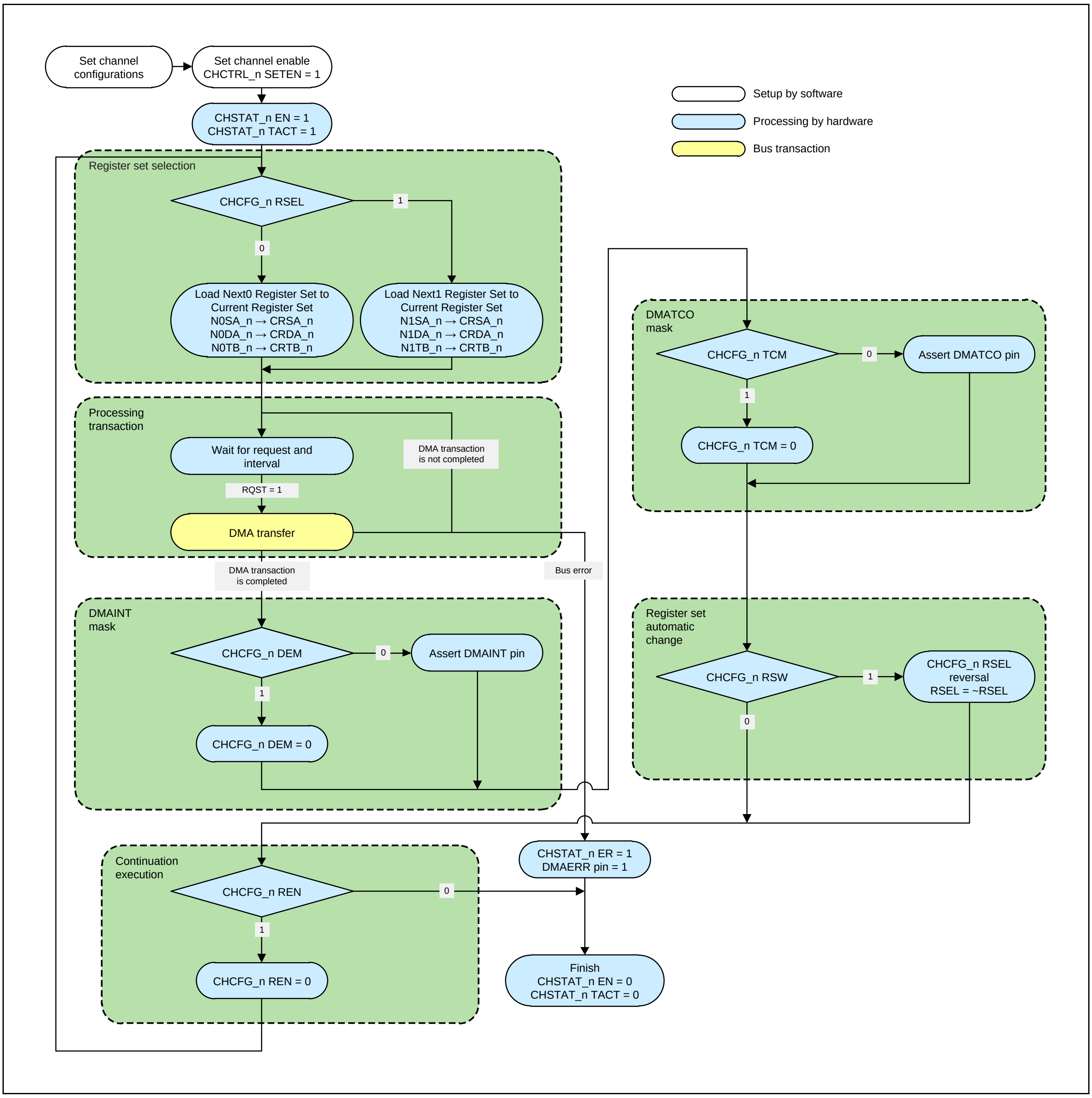


Figure 32B.15 Register Mode Operation Flow

Description of the register mode operation flow:

1. Channel setting
   The Next0 Register Set or Next1 Register Set register is set (transfer-destination address, transfer-source address, and total number of bytes to be transferred). In addition, the FIFO channel, volume of transfer data, and other items are set for USB control that is used for the Channel Register Set register.
   No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.
   (see **Section 32B.9.13, DMA Transfer**).

2. Register set selection
   When 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits in the CHSTAT_n register are set to 1. As a result, the values set by the Next Register Set register selected by the RSEL bit in the CHCFG_n register are loaded to the Current Register Set register.

3. DMA transaction
   A DMA transaction is executed based on the values that are set. For details about transfer, **Section 32B.9.13, DMA Transfer**.

4. The USBFDMAmn masking
   The USBFDMAmn interrupt is masked depending on the value of the DEM bit in the CHCFG_n register. If DEM = 1, the USBFDMAmn interrupt is masked, and the DEM bit is automatically cleared to 0.

5. DMATC masking
   DMATC from DMAC control to USB control is masked depending on the value of the TCM bit in the CHCFG_n register. If TCM = 1, DMATC is masked, and the TCM bit is automatically cleared to 0.

6. Automatic register set switchover
   Whether the current Next register set is to be switched to the other Next register set is determined by the value of the RSW bit in the CHCFG_n register.

7. Continuation of execution
   Whether to continue DMA transfer is determined by the value of the REN bit in the CHCFG_n register. If REN = 0, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC operation stops. If REN = 1, DMAC operation continues, and the REN bit is automatically cleared to 0.

(b) Register mode setting

- Register mode setting

The register set to be processed is selected.

Table 32B.37 Register Mode Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0	0	Processes Next0 Register Set.
	1	Processes Next1 Register Set.

- USBFDMAmn masking

The USBFDMAmn interrupt can be masked.

Table 32B.38 USBFDMAmn Mask Setting

DEM (CHCFG_n)	Description
0	Asserts the USBFDMAmn interrupt when the DMA transaction is completed.
1	Does not assert the USBFDMAmn interrupt even when the DMA transaction is completed. After the DMA transaction is completed, the DEM bit is cleared to 0.

- DMATC mask setting

DMATC from DMAC control to USB control can be masked.

Table 32B.39 DMATC Mask Setting

DEM (CHCFG_n)	Description
0	Asserts DMATC when the DMA transaction is completed.
1	Does not assert DMATC even when the DMA transaction is completed. After the DMA transaction is completed, the TCM bit is cleared to 0.

- Automatic transaction execution for a register set

After a DMA transaction finishes, another DMA transaction can be executed.

Table 32B.40 Automatic Execution Setting for a Register Set

REN (CHCFG_n)	Behavior	Remarks
0	The EN bit is cleared and DMA operation is terminated when the DMA transaction for the register set that is set by RSEL finishes.	Use this setting to execute a DMA transaction only once.
1	After a DMA transaction finishes, DMA transfer of the contents of the next register set continues. The REN bit is cleared to 0 when continuous transfer is successful.	Use this setting to continue processing of register set contents.

- Automatic register set switchover setting

After a DMA transaction finishes, the next register set to be processed can be switched.

Table 32B.41 Automatic Execution Setting for a Register Set

RSW (CHCFG_n)	Behavior	Remarks
0	The register set is not switched when a DMA transaction finishes.	Use this setting to use only one register set.
1	When REN = 1 and a DMA transaction finishes, the RSEL setting is automatically reversed to select the other register set.	Use this setting to switch the register set.

(c) Example of setting the register mode

- Example of setting the register mode when using only the Next0 register set

Table 32B.42 Register Mode Setting Example

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	0 (not masked)	0 (not masked)	0 (not switched)	0 (continuous execution disabled)

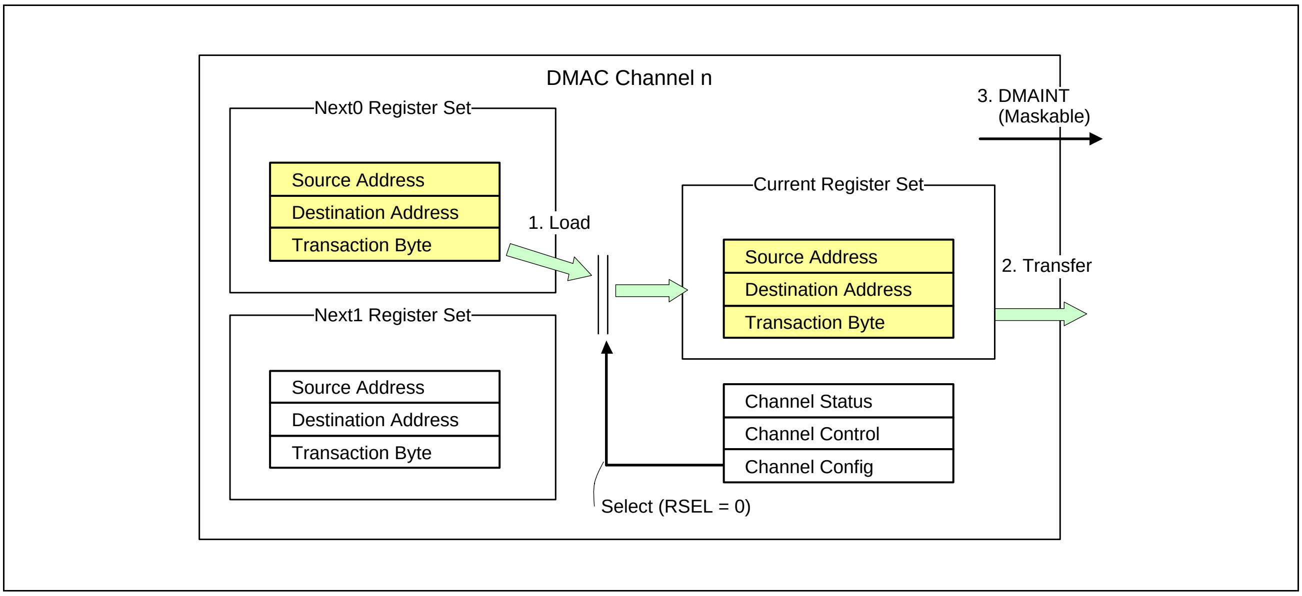


Figure 32B.16 Register Mode Setting Example 1

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
4. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
5. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

- Example of setting the register mode when using two register sets continuously

Table 32B.43 Automatic Register Set processing Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	1 (masked)	0 (not masked)	1 (switched)	1 (continuous execution enabled)

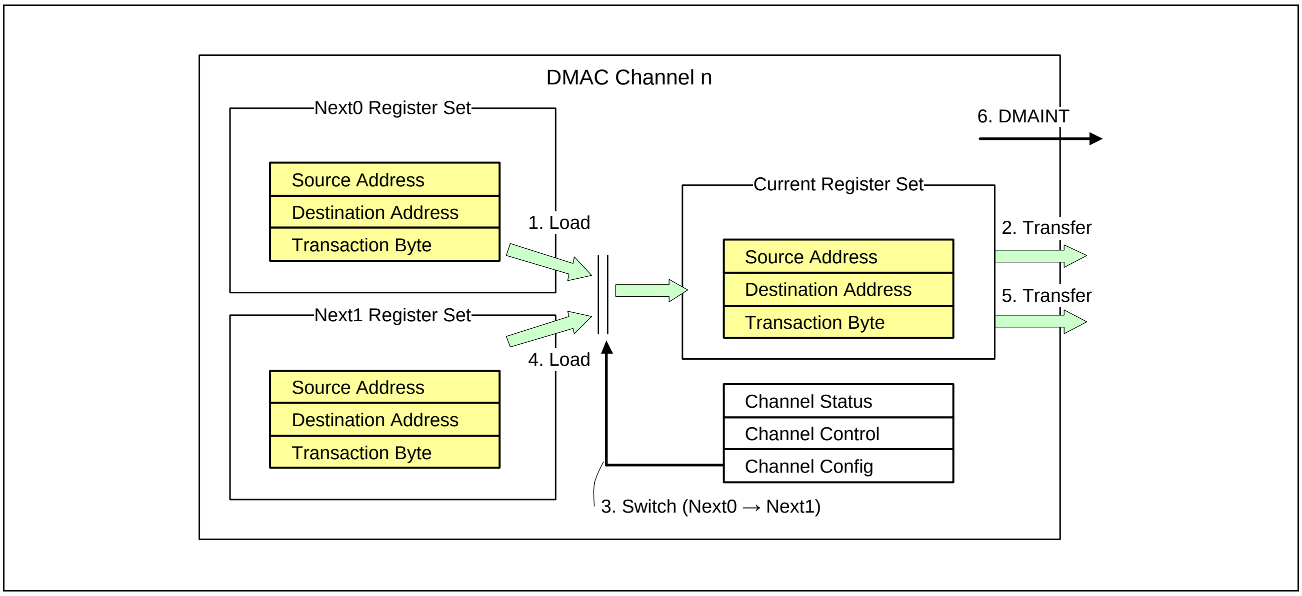


Figure 32B.17 Register Mode Setting Example 2

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 1, the USBFDMAmn interrupt is not asserted after the DMA transaction finishes.
4. Because the REN bit in the CHCFG_n register is 1, operation continues. The REN bit is automatically cleared to 0.
5. Because the RSW bit in the CHCFG_n register is 1, the next register set to be processed is switched (RSEL = 0 to 1).
6. The contents of Next1 Register Set are loaded to Current Register Set.
7. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
8. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
9. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
10. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

(2) Link mode

In link mode, this module reads the value set in a descriptor placed in an external storage area to execute a DMA transaction. In DMAC, there are Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers for each channel. The Next Link Address (NXLA_n) register is used to set the address of the descriptor to be read the next time. The Current Link Address (CRLA_n) register is used to display the descriptor address for the current DMA transaction.

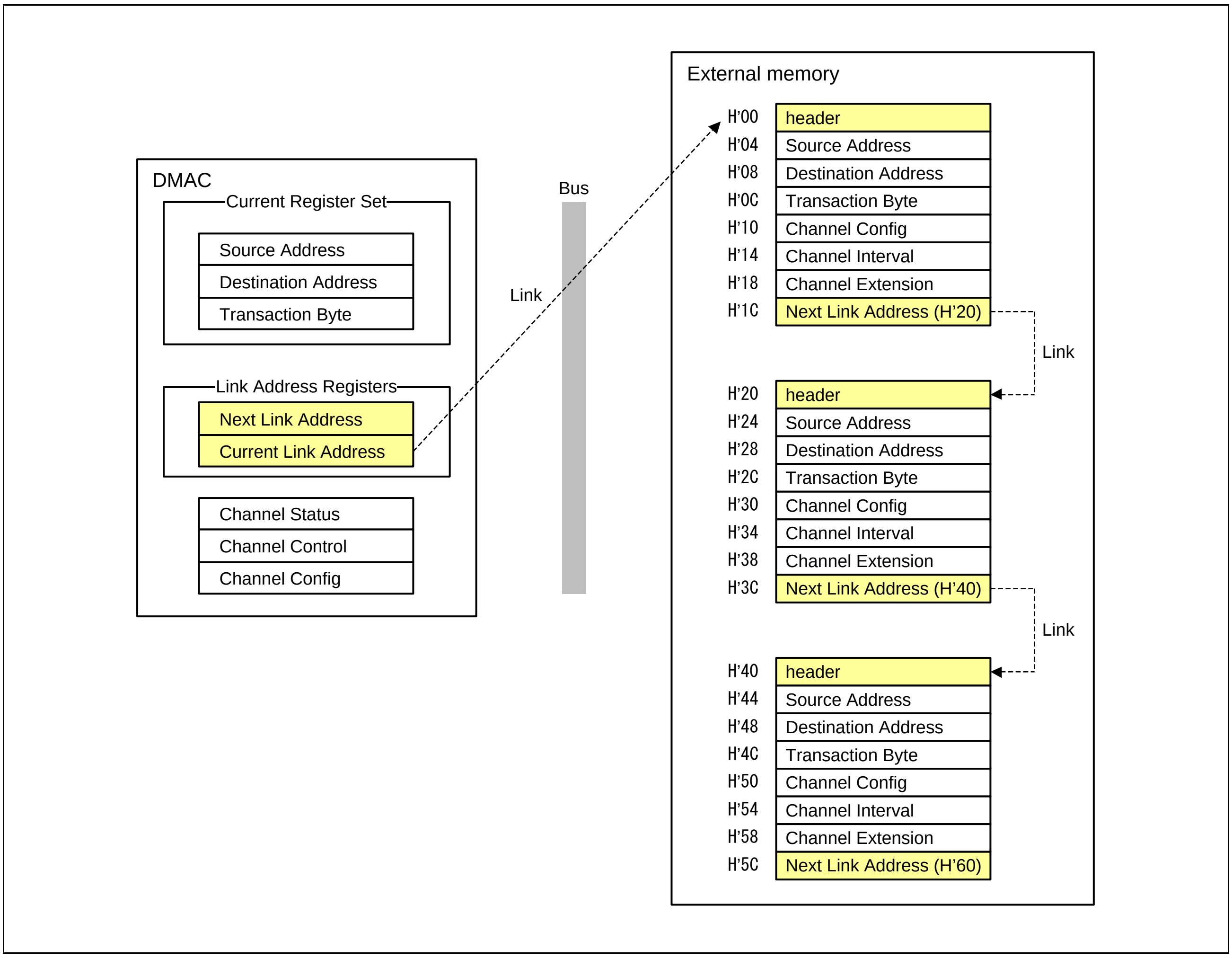


Figure 32B.18 Overview of Link Mode

(a) Operation flow of link mode

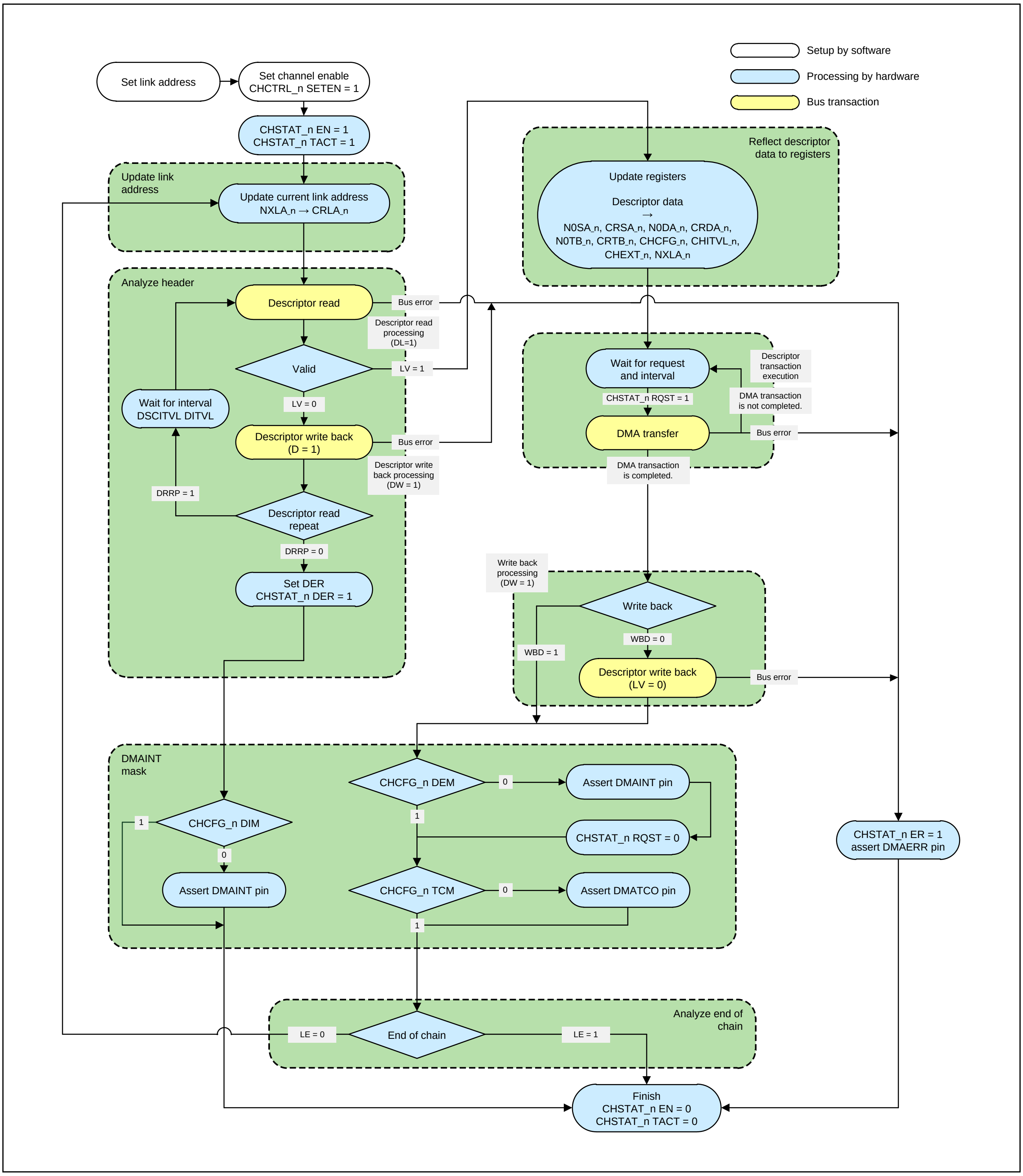


Figure 32B.19 Link Mode Operation Flow

Description of the link mode operation flow:

1. Channel setting
   The beginning address of the link destination is set in the NXLA_n register.

2. Link address updating
   If 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits of the CHSTAT_n register are set to 1. As a result, the address set in the NXLA_n register is loaded to the CRLA_n register.

3. Descriptor reading and header judgment
   A read of the descriptor starts, and DMAC checks the contents of header. If LV = 0, this module writes 1 back to the D bit of header. After that, if the DRRP bit in the CHCFG_n register is 1, this module waits for the time intervals set by the DSCITVL register, and then reads the same descriptor again. If DRRP = 0, the DER bit in the CHSTAT_n register is set to 1, and this module is placed in the end state (both EN and TACT bits in the CHSTAT_n register are 0). At this time, if the DIM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.

4. Descriptor setting
   If LV = 1, the data read from the descriptor is loaded to Current Register Set and Channel Register Set. In addition, the next link target is loaded to the NXLA_n register.

5. DMA transaction
   A DMA transaction is executed based on the values that are set. For details about transfer, see **Section 32B.9.13, DMA Transfer**.

6. Header write-back
   If WBD of header is 0, DMAC writes LV = 0 to the header area.

7. USBFDMAmn masking
   If the DEM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.

8. DMATC masking
   If the TCM bit in the CHCFG_n register is 0, this module asserts DMATC.

9. Link end judgment
   If LE of header is 1, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC terminates operation. If LE is 0, this module updates Current Register Set, and then restarts reading the next descriptor.

(b) Register setting

- Link mode setting

To use link mode, set the DMS bit in the CHCFG_n register to 1.

Table 32B.44 Link Mode Setting

DMS (CHCFG_n)	Description
1	This module operates in link mode. The setting of this bit cannot be changed by using a descriptor.

- Link address setting

The Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers are used to indicate a link target. Before starting link mode, set the link target in the NXLA_n register.
After reading a descriptor, this module updates the NXLA_n register to the next link. Note that the CRLA_n register indicates the address of the link target that is being executed.

Table 32B.45 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n)	This register is used to set and display the next link target. Before starting link mode, set the address of the link target in this register.
Current Link Address Register (CRLA_n)	This register is used to display the link target that is being executed. This is a read-only register.

(c) Descriptor setting

DMAC supports multiple descriptor formats.

A switchover between formats is specified by using the DSCFM field of bits [31:28] of the 1st word (header) of the descriptor.

The following table shows the relationship between DSCFM values and descriptor formats.

Table 32B.46 Descriptor Formats

DSCFM	Descriptor Size	Next Link Address	Channel Extension	Channel Interval	Channel Config	Transaction Size	Destination Address	Source Address	header
3	4 words	✓	— (reload)	— (reload)	— (reload)	— (header)	✓	✓	✓ (with STS)
1	8 words	✓	✓	✓	✓	✓	✓	✓	✓ (no STS)
Other than the above	If DSCFM is set to a value that is not 1 or 3, operation cannot be guaranteed. Make sure that DSCFM is set to 1 or 3.								

Table 32B.47 Explanation of the Marks in **Table 32B.46**

Field	Mark	Description	Remarks
Header	✓ (with STS)	The STS field of bits [15:0] in the header is valid. The value set in the STS field is used as the total number of transfer bytes (Transaction Size).	—
	✓ (no STS)	The STS field of bits [15:0] in the header is invalid. The value of "Transaction Size" in the descriptor is used as the total number of bytes.	
Source Address	✓	Specify the source address.	—
Destination Address	✓	Specify the destination address.	—
Transaction Size	✓	Specify the transaction size.	—
	— (header)	Omit the transaction size. The value set in the STS field is used as the total number of transfer bytes (Transaction Size)	Because the STS field is of 16 bits, a maximum of 65,535 bytes can be set.
Channel Config Channel Interval Channel Extension	✓	Specify Channel Config, Channel Interval, and Channel Extension.	—
	— (reload)	Omit Channel Config, Channel Interval, and Channel Extension. The previous settings (the values of the CHCFG_n, CHITVL_n, and CHEXT_n registers of the last time) are inherited.	—
Next Link Address	✓	Specify the next descriptor address (Next Link Address) to be read after DMA transfer of the descriptor.	—

DMAC sequentially interprets data read from descriptors. If the number of words specified for DSCFM is less than 8, place the data of descriptors that are indicated by "✓" in **Table 32B.46** on memory.

No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.

Table 32B.48 Example of Placing Descriptors

DSCFM	Address							
	Link Address + H'1C	Link Address + H'18	Link Address + H'14	Link Address + H'10	Link Address + H'0C	Link Address + H'08	Link Address + H'04	Link Address + H'00
H'3	—	—	—	—	Next Link Address	Destination Address	Source Address	header
H'1	Next Link Address	Extension	Interval	Config	Transaction Byte	Destination Address	Source Address	header

- header

The header area provides the descriptor status and other information as shown below.

DMAC reads this area before DMA transfer in link mode starts. After a DMA transaction terminates, DMAC writes the transfer status back to this area.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSCFM				—	WBD	LE	LV	D	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W																

If the WBD bit is 0 when DMAC reads the header, DMAC writes data back to this 1-byte area after the DMA transaction in accordance with the descriptor ends. At this time, DMAC writes 0 back to the LV bit. For other bits, DMAC writes the values read from the header.

If the LV bit in the header is 0 when DMAC reads the header, DMAC writes H'80 back to this 1-byte area.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STS															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W																

Figure 32B.20 Header Area

Table 32B.49 Header Area (1/2)

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DSCFM			Descriptor Format Specifies the descriptor format (length and combination of descriptors). For details see **Table 32B.46**.
27	—			Reserved area. Set 0.
26	WBD			Wrazite Back Disable Masks a write-back operation for the LV bit. If this bit is 1, DMAC does not perform a write-back operation. 0: Writes 0 back to the LV bit. 1: Does not perform a write-back operation for the LV bit.
25	LE			Link End Indicates that the link will end with the DMA transaction for this descriptor. Set this bit to 1 to indicate the end of link. 0: The link continues. 1: The link ends.

Table 32B.49 Header Area (2/2)

Bit	Bit Name	Initial Value	R/W	Description
24	LV			Link Valid Indicates that this descriptor is valid. If WBD = 0, after DMAC executes the DMA transaction written in the descriptor, DMAC writes 0 to this bit. When header is set, set 1 to this bit. 0: This descriptor is invalid. 1: This descriptor is enable.
23	D			Descriptor Error Indicates a descriptor access error. If LV is 0 when the descriptor is read, DMAC writes 1 back to this bit. 0: A descriptor error has not occurred. 1: LV was 0 when the descriptor was read.
22 to 16	—			Reserved area. Set 0.
15 to 0	STS			Short Transaction Size If DSCFM is 3, the transaction size is set (in bytes). The maximum number of transfer bytes that can be set is 65,535. If DSCFM is 3, Do not set 0 for STS. If 0 is set, operation cannot be guaranteed.

If descriptors are added sequentially while DMAC is operating, the access that the CPU sets 0 to the LV bit and the access that DMAC writes 1 back to the D bit might contend with each other. If this contention occurs, prior-written data is overwritten by latter-written data.

To prevent this problem from occurring, DMAC performs a write-back operation for the D bit in a byte-write manner. Therefore, the CPU must also set LV to 1 in a byte-write manner. Because the byte lanes for the D and LV bits are different, by writing data to different areas, occurrence of this problem can be prevented.

- Settings of descriptors other than header

The specifications of data of the descriptors other than header are the same as the specifications of internal registers.

- Settings specified when descriptors are accessed

The MHPROT pin output can be set for the LWPR and LDPR fields of the DCTRL register when descriptors are accessed. Set it according to the access target in which descriptors are deployed.

- Descriptor areas and DMA transfer areas

The following provides an overview of the descriptor areas and DMA transfer areas that are accessed by DMAC.

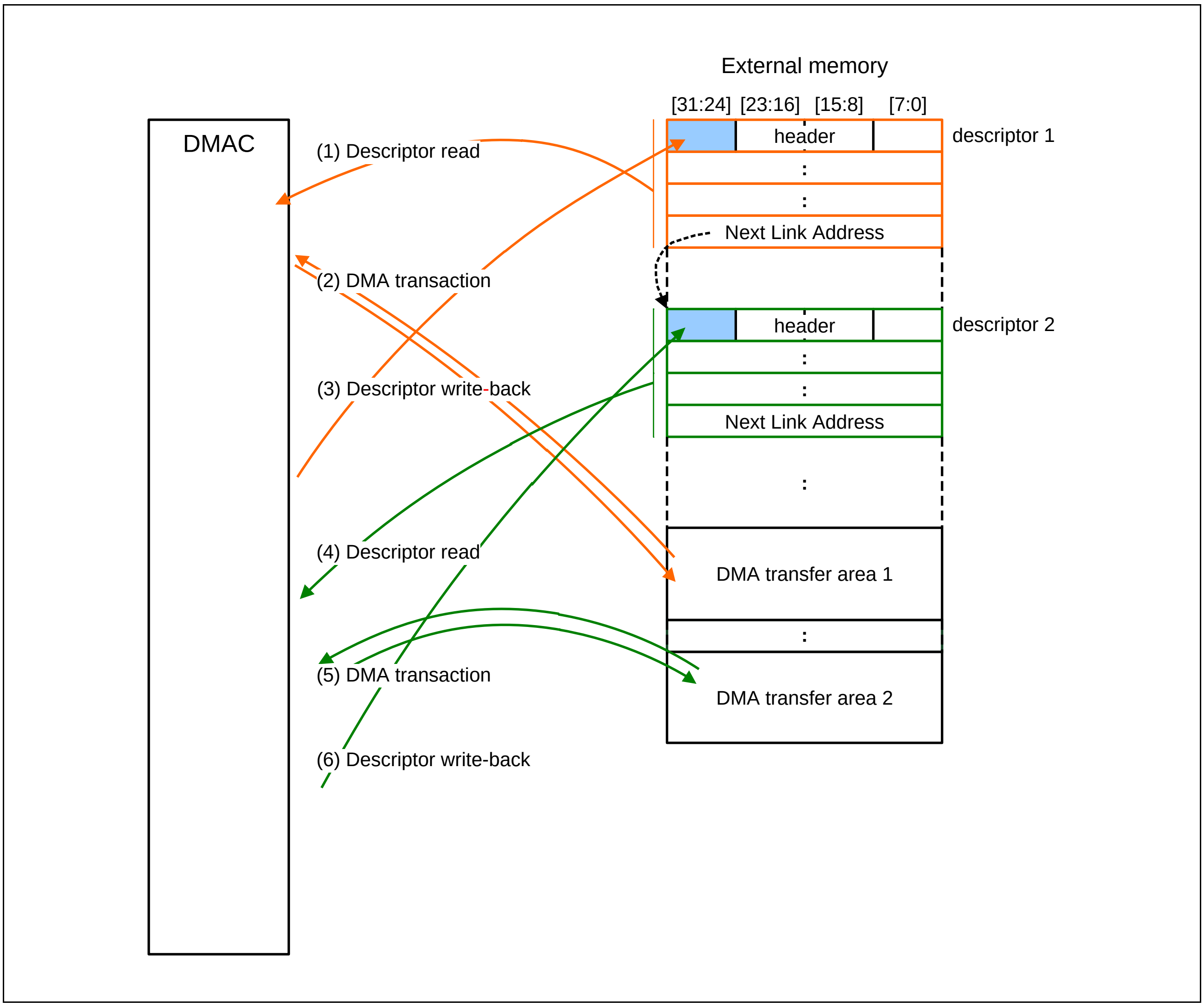


Figure 32B.21 Overview of Descriptor Areas and DMA Transfer Areas

(1) Descriptor read
DMAC loads a value from the internal NXLA_n register to the CRLA_n register, and then reads a descriptor from the external memory space indicated by the CRLA_n register (descriptor 1).

(2) DMA transfer
If the LV bit of the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(3) Descriptor write-back
After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 1. For the LV field, 0 is written back. For other fields, the values read in (1) are written back on a byte-size basis.

(4) Descriptor read
If the value of the LE bit in the header descriptor that was read previously (in (1)) is 0, DMAC reads the next descriptor from the address (descriptor2) indicated by Next Link Address in the current descriptor.

(5) DMA transfer

If the LV bit in the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(6) Descriptor write-back

After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 2. For the LV field, 0 is written back. For other fields, the values read in (4) are written back on a byte-size basis.

(Steps (4) to (6) are repeated.)

If LE = 1 and WBD = 0 in header, DMAC performs a DMA transfer with the descriptor settings, writes 0 back to the LV bit in header, and then terminates processing.

If LE = 1 and WBD = 1 in header, DMAC performs a DMA transfer with the descriptor settings, and then terminates processing (without performing write-back).

If LV = 0 in header, DMAC writes 1 back to the D bit in header, and then, if the DRRP bit in the CHCFG_n register is 1, DMAC waits for the number of intervals specified by the DITVL field of the DSCITVL_n register, and then reads the descriptor again. If DRRP = 0, DMAC terminates processing.

- Notes on descriptors
  - In link mode, settings can be changed by reading descriptors. However, it is impossible to synchronize the setting change times and hardware requests. Therefore, to use hardware requests, before setting the SETEN bit in the CHCTRL_n register, set the AM, LVL, HIEN, LOEN, and SEL bits in the CHCFG_n register. Note that these setting bits must not be changed in descriptors.
  - Descriptors cannot be used to change the DMS field in the CHCFG_n register (DMAC is always placed in link mode). Although descriptors can be used to change the settings of the REN, RSW, and RSEL fields in the CHCFG_n register, changes of those fields do not affect operation.
  - The descriptor can be initialized by overwriting the memory area corresponding to the descriptor you intend to initialize while the DMAC is not operating. The DMAC determines whether or not the descriptor is valid by referring to the DSCFM field and LV bit in the header. Accordingly, set the areas in memory corresponding to the DSCFM field and LV bit to 1 or 3, and to 1, respectively, before enabling DMAC operation.
  - To set the next descriptor on memory while DMAC is operating, make sure that 1 is written to the LV bit after the descriptors subsequent to header (Source Address, Destination Address, ..., Next Link Address) are set. If this is not performed and descriptor setting by the CPU and descriptor reading by DMAC contend, DMAC performs a DMA transfer using the previous values of those descriptors (Source Address, Destination Address, ..., Next Link Address).
  - To leave the write-back information for the D bit of header, make sure that 1 is written to the LV bit of header in a byte-access manner.

(d) Link configuration example

In link mode, descriptors can be configured as shown below.

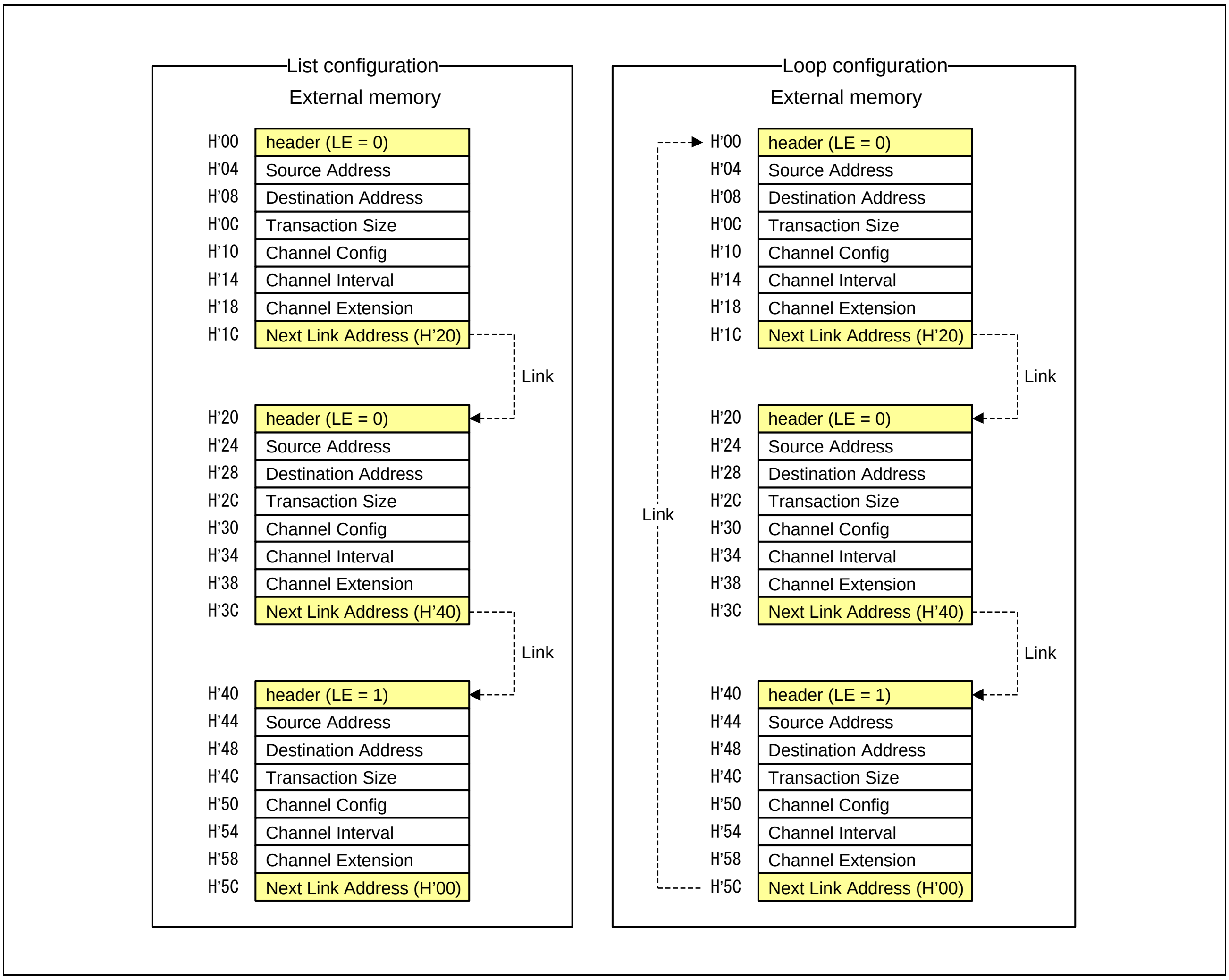


Figure 32B.22 Link Mode Configuration Example

- List configuration

  The link ends by setting 1 for the LE bit in the header of the last descriptor.

- Loop configuration

  A loop of descriptors can be created by setting the link target of the last descriptor to the address of the first descriptor. To end the loop, change the value of the LE bit of header to 1 or use the transfer interrupt procedure.

32B.9.12.2 Write only mode

Write-only mode is enabled by setting 1 for the WONLY bit in the CHCFG_n register.

Table 32B.50 Write-Only Mode Setting

WONLY (CHCFG)	Mode	Description
0	Normal mode	A DMA transfer is performed using the values set in Next Register Set.
1	Write-only mode	A DMA write transfer is performed without performing a DMA read transfer.

In write-only mode, no DMA read transfer is performed (note that descriptors are read in the same way as in normal mode). In register mode, the values set in the NxSA_n register (if RSEL = 0, x = 0; if RSEL = 1, x = 1) are used as write data. In link mode, the values of the SA fields of descriptors are used as write data.

Use this mode to, for example, initialize the memory area.

32B.9.13 DMA Transfer

This chapter describes the basic operation of DMA transfer.

32B.9.13.1 Transfer modes

DMAC supports only single transfer mode.

Upon receiving a DMA transfer request from USB control, DMAC executes a single DMA transfer on the side (source or destination) indicated by the REQD bit in the CHCFG_n register. DMAC then asserts internal DMA permission from internal USB control to DMAC control. DMAC performs a single transfer each time a transfer is received. DMAC continues this operation by the transfer size loaded to the CRTB_n register (arbitration between channels is performed for each DMA transfer).

The timing of internal DMA permission from internal USB control to DMAC control differs depending on the setting of the REQD bit in the CHCFG_n register and the setting of the transfer size (DDS[2:0] and SDS[2:0] in the CHCFG_n register). For details, see **Section 32B.9.13.7, Operational difference depending on the transfer size**.

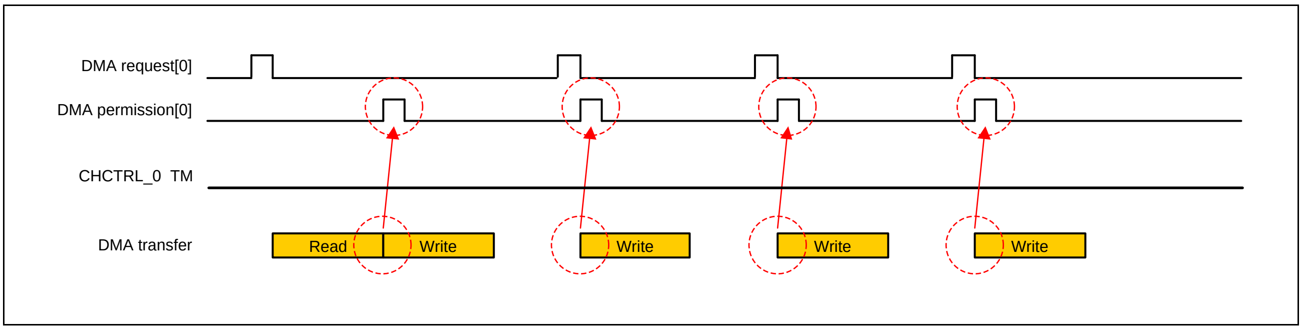


Figure 32B.23 Single Transfer Mode (REQD = 1, SDS > DDS)

32B.9.13.2 DMA channel priority control

DMAC supports fixed-priority mode and round-robin mode as methods of arbitration between channels. The mode is selected by using the PR bit in the DCTRL register. If the PR bit is 0, fixed-priority mode is selected. If the PR bit is 1, round-robin mode is selected.

Table 32B.51 Priority Control Setting

Mode	PR (DCTRL)	Description	Remarks
Fixed-priority	0	Controls requests based on fixed priority (CH0 > CH1 > ...).	Use this mode if channels have priority.
Round-robin	1	Controls requests in a round-robin manner.	Use this mode to execute requests equally.

(1) Fixed-priority mode

In fixed-priority mode, a fixed priority level is assigned to each channel as shown below.

(High) CH0 > CH1 (Low)

If DMA transfer requests simultaneously occur over multiple channels, the DMA transfer over the channel whose number is smallest is performed first.

The transfer over channel 0 is performed first. However, while a transfer switches to another transfer over channel 0, a transfer over the channel with the next highest priority level is performed in order to increase the bus usage rate.

(2) Round-Robin Mode

In round-robin mode, each time a transfer over a channel is received, the priority level of the channel that was used for the previous transfer is changed to the lowest level.

In the status immediately after the mode is reset, channels are assigned priority levels in the same way as fixed-priority mode as shown below.

(High) CH0 > CH1 (Low)

In this status, if a transfer request for DMA channel 0 does not occur and a transfer request for DMA channel 1 occurs, the transfer over DMA channel 1 is performed. When the transfer finishes, the channel priority is changed as follows.

(High) CH1 > CH0 (Low)

The following shows an example of DMA transfer in round-robin mode.

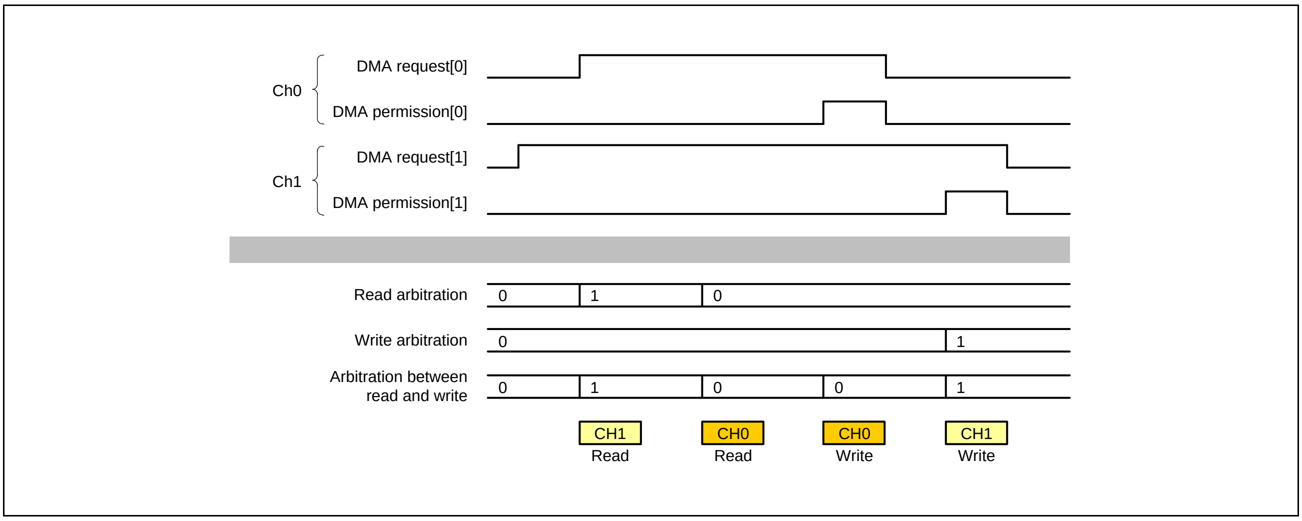


Figure 32B.24 Example of Operation in Round-Robin Mode (with 4 channels, REQD = 1)

DMAC internally performs arbitration between read channels and arbitration between write channels, further performs arbitration between the arbitration results, and then issues bus access.

32B.9.13.3 Forced sweeping request

When a forced sweeping request is entered, DMAC transfers the data that is left untransferred in the buffer to the destination address. After the sweep processing finishes, DMAC continues DMA transfer.

The following provides notes on forced sweeping requests:

(1) If a forced sweeping request and a transfer request from USB control contend, DMAC performs forced sweeping first, and then performs a DMA transfer.

(2) If the USB-control-side system is the destination (REQD bit in the CHCFG_n register is 1) register REQD = 1, buffer overflow or another error might occur on the destination unit because data is transferred although no DMA transfer request is made on the USB control side. Therefore, the specifications physically prohibit DMAC from using forced sweeping if REQD = 1.

(3) The difference from ordinary sweeping mode described in **Section 32B.9.13.8(3)(b), Transfer suspension (buffer sweeping enabled: SBE = 1)** (EN is cleared by setting 1 for the SBE bit in the CHCFG_n register) is as follows: DMAC stops operation after writing data in the buffer in ordinary sweeping mode, whereas DMAC can continue DMA transfer after sweeping the buffer in forced sweeping mode.

(1) Forced software sweeping request

The SETSSWPRQ bit in the CHCTRL_n register determines whether software-based forced sweeping requests can be used. To perform a forced sweeping request, write 1 to the SETSSWPRQ bit. DMAC then outputs the data in the buffer to the destination.

32B.9.13.4 DMA transfer completion interrupt (USBFDMAmn)

USBFDMAmn (m, n = 0, 1) is an interrupt signal that indicates termination of a DMA transaction.

If a transfer for the total number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response, the END bit in the CHSTAT_n register is set to 1. At this time, if the DEM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn (m, n = 0, 1) interrupt.
(If a write-back operation is performed in link mode, the interrupt is generated after the write-back operation finishes.)

In link mode, when the DRRP bit in the CHCFG_n register is 0, if the LV bit of the header of the descriptor that is read is 0, the DER bit in the CHSTAT_n register is set to 1. At this time, if the DIM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn interrupt.

Use this signal to detect a transfer completion interrupt performed by the interrupt controller.

Table 32B.52 USBFDMAmn Assertion Conditions

Cause	Condition	INT_DMA[n] mask signal
DMA transaction ended	A transfer of data by the number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response (if a write-back operation is performed in link mode, after the operation finishes)	DEM bit in the CHCFG_n register
Descriptor was invalid	LV of header of the descriptor that is read is 0 when DRRP and DIM in the CHCFG_n register are both 0 in link mode	DIM bit in the CHCFG_n register

32B.9.13.5 DMA error interrupt (USBFDMAERRm)

If an error response is received for DMA transfer or descriptor access, this module stops transfer, assuming that an error occurred. When an error response is received, the EN bit in the CHSTAT_n register for channel n that is being used for transfer is cleared to 0, and the ER bit is set to 1 (n = 1, 0). Also, the USBFDMAERRm interrupt is asserted.

The USBFDMAERRm signal cannot be masked.

For a sequence of transfers for which an error occurred, data integrity cannot be guaranteed. Always use the following procedure to restart the transfer sequence from the beginning.

1. Set the SWRST bit in the CHCTRL_n register to 1.
2. Reset each register.

32B.9.13.6 Interval Count Function

The execution interval of a DMA transfer can be adjusted by using the ITVL field in the CHITVL_n register. This function prevents DMAC from continuously occupying the bus. If this function is enabled, DMAC does not perform a DMA transfer for the next request until the counter value becomes 0.

The following shows an operation example.

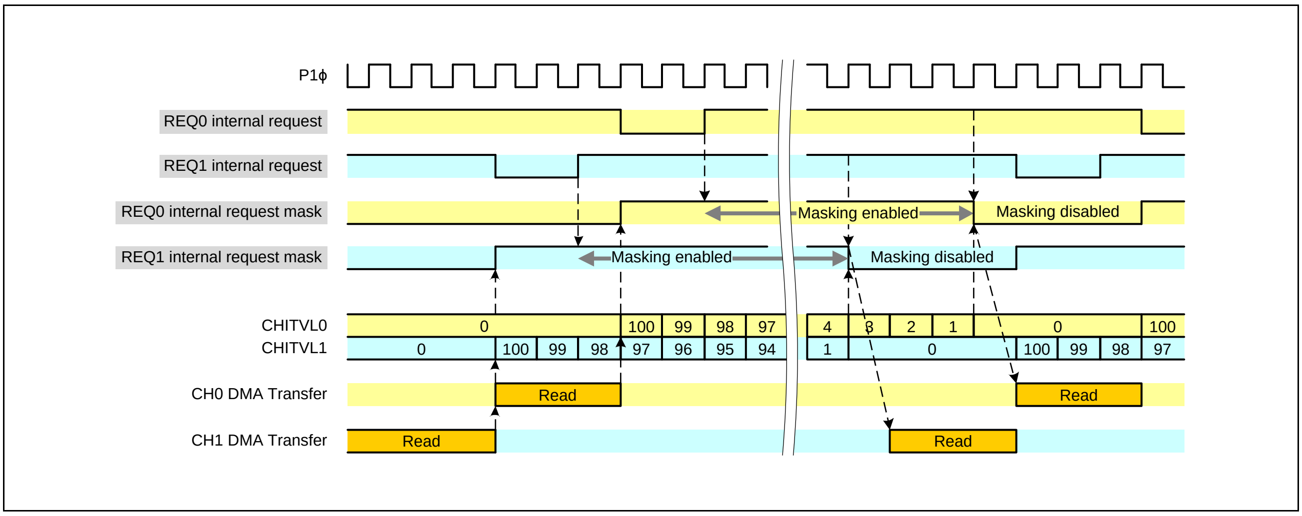


Figure 32B.25 Example of Counting Intervals (REQD = 0, SDS < DDS)

An interval is inserted after a transfer is performed on the side specified by the REQD bit in the CHCFG_n register. The following shows how the REQD, SDS, and DDS values of the CHCFG_n register are related with the interval.

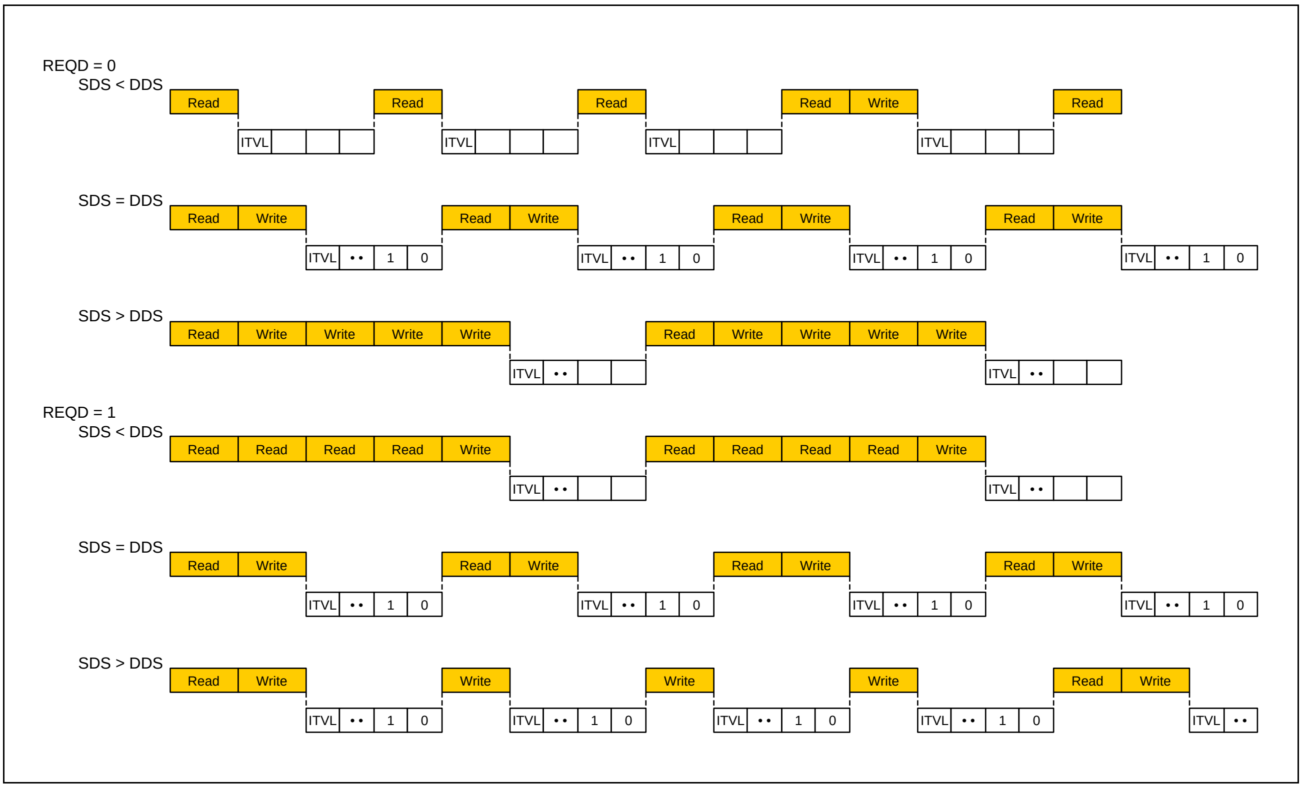


Figure 32B.26 DMA Transfer Settings and Interval Count

32B.9.13.7 Operational difference depending on the transfer size

(1) If the transfer size on the source side is smaller

When reading of as much data as the destination data size finishes, a write to the destination starts.

The following figure is an example of the timing chart in the case where the source is an 8-bit field and the destination is a 32-bit field (SDS = 0 and DDS = 2 in the CHCFG_n register) (when the rising edge is detected).

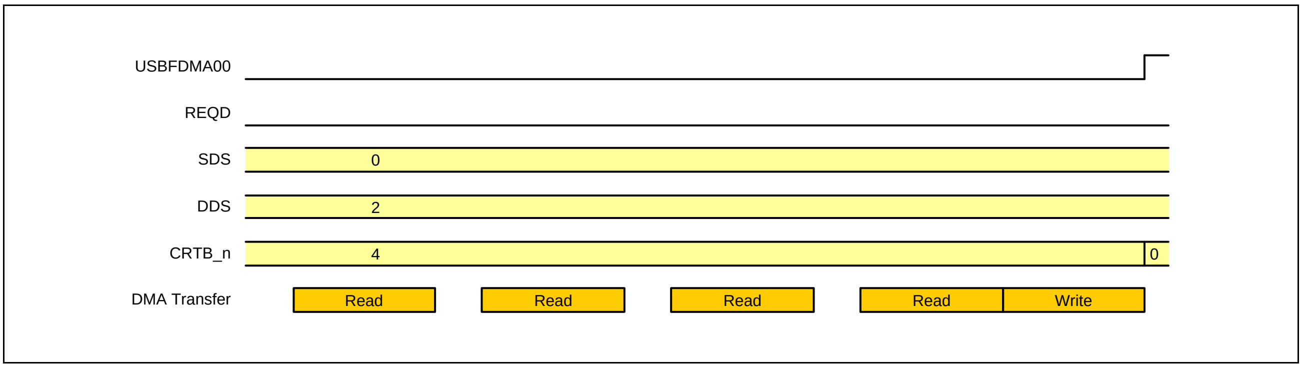


Figure 32B.27 Example of Timing Chart in the Case Where the Source is Smaller
(LVL = 0, HIEN = 1, REQD = 0, and SDS < DDS)

(2) If the transfer size on the destination side is smaller

Because the source side is larger than the destination side, two or more destination write operations occur for one source read operation. The following is an example of the timing chart in the case where the source is a 64-bit field and the destination is a 16-bit field (SDS = 3 and DDS = 1 in the CHCFG_n register) (when the rising edge is detected).

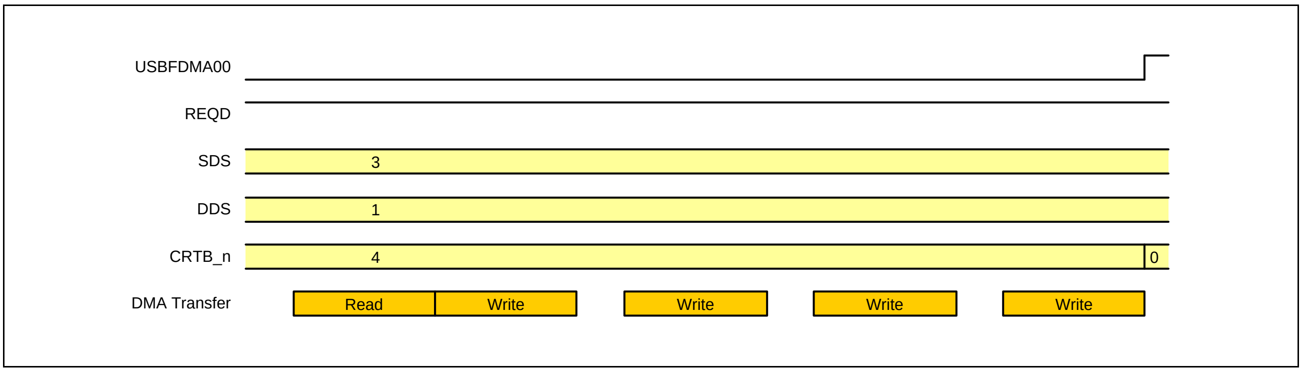


Figure 32B.28 Example of Timing Chart in the Case Where the Destination Is Smaller
(LVL = 0, HIEN = 1, REQD = 1, and SDS > DDS in the CHCFG_n Register)

(3) If the Source and Destination Transfer Sizes Are the Same

Each time a DMA transfer request is detected, a source read operation and a destination write operation occur.

The following is an example of the timing chart in the case where the source and destination are 8-bit fields (SDS = 0 and DDS = 0 in the CHCFG_n register) (when the rising edge is detected).

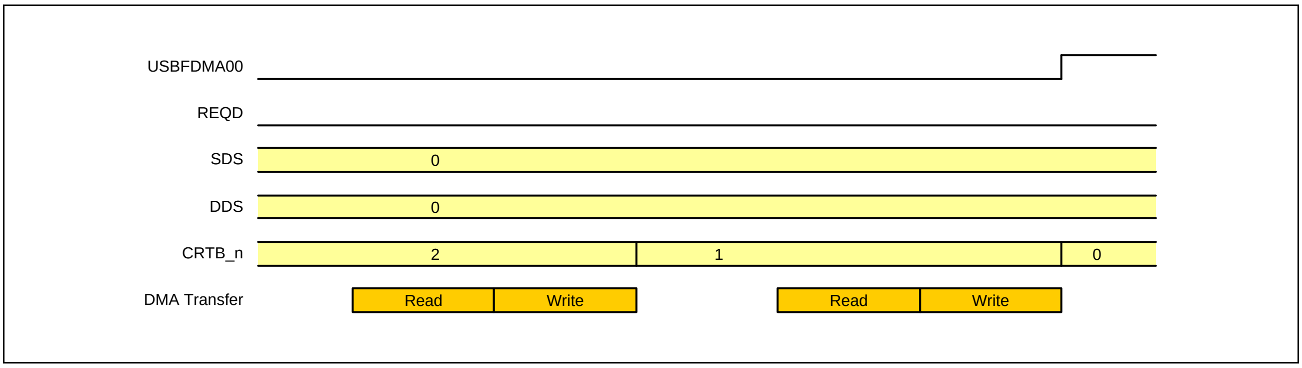


Figure 32B.29 Example of Timing Chart in the Case where the Source and Destination Sizes are the Same (LVL = 0, HIEN = 1, REQD = 0, and SDS = DDS in the CHCFG_n Register)

32B.9.13.8 Transfer state

The CHSTAT_n register indicates the transfer state of a channel.

(1) Transfer state

The TACT bit in the CHSTAT_n register indicates that channel n is operating. When 1 is written to the SETEN bit in the CHCTRL_n register, the TACT bit is set to 1. The TACT bit continues to be 1 while DMAC is accessing a descriptor or waiting for a DMA request.

The TACT bit is cleared when the EN bit in the CHSTAT_n register is cleared (for details about the conditions in which the EN bit is cleared, see **Section 32B.4.1, Current Source Address Register**) and the DMA transfer ends.

If the EN bit is not cleared when the DMA transaction ends (for example, when the REN bit of the CHCFG_n register is 1 in register mode or when DMAC accesses the next descriptor in link mode), the TACT bit is not cleared.

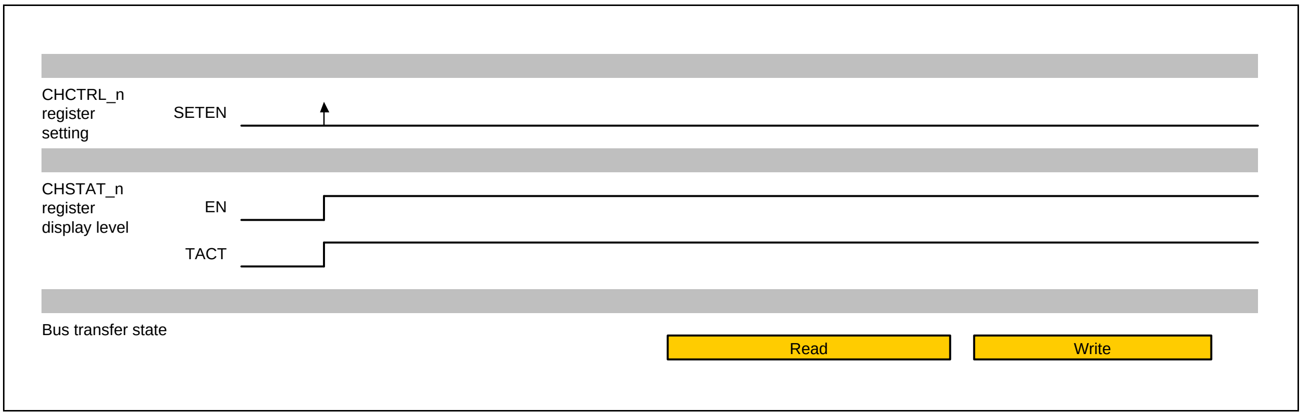


Figure 32B.30 DMAC State Example 1 (Hardware Request)

(2) Suspension

The SETSUS bit in the CHCTRL_n register can be used to suspend a DMA transfer. If suspension of a DMA transfer is attempted when a bus cycle is running, the DMA transfer is suspended after the bus cycle finishes. The suspended transfer can be resumed by writing 1 to the CLRSUS bit in the CHCTRL_n register.

To check whether a DMA transfer is suspended, after setting the SETSUS bit in the CHCTRL_n register, check the SUS bit in the CHSTAT_n register or the SUS bit for the relevant channel in the DSTAT_SUS register. If The SUS bit is 1, the DMA transfer is currently suspended.

(3) Transfer suspension

By writing 1 to the CLREN bit in the CHCTRL_n register during a DMA transaction, the DMA transaction of the channel can be suspended. As the post-processing for suspension, the SBE bit in the CHCFG_n register can be used to select whether to sweep the data remaining in the buffer when the transaction is suspended. The default is SBE = 0 (do not sweep remaining data).

If the sweep mode is enabled and a transfer is suspended by setting 1 for the CLREN bit in the CHCTRL_n register, DMAC sweeps any data remaining in the buffer and stops operation.

(a) Transfer suspension (buffer sweeping disabled: SBE = 0)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer and then stops. The timing of stoppage depends on the value set for the REQD bit. After DMAC stops, write 1 to the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.

Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.

If DMA transfer is suspended before it is completed, the USBFDMAmn interrupt is not asserted.

If the REQD bit in the CHCFG_n register is 0, DMAC stops when the next read operation is completed. However, if data that can be written exists in the buffer, DMAC writes the data and then stops.

If the REQD bit in the CHCFG_n register is 1, DMAC stops when the next read operation is completed.

(b) Transfer suspension (buffer sweeping enabled: SBE = 1)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer. If the REQD bit in the CHCFG_n register is 0, DMAC sweeps (writes) the already read data, and then stops DMA transfer. If the REQD bit is 1, sweep mode cannot be used physically.

After DMAC stops, set the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.

Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.

(c) How to confirm deactivation of the channel

Even when the EN bit of the CHSTAT_n register is cleared to 0 by writing 1 to the CLREN bit in the CHCTRL_n register, if a transfer has already been executed over the bus, DMAC cannot immediately stop. To check whether DMAC has stopped completely, check the EN and TACT bits in the CHSTAT_n register. If both bits are 0, DMAC has stopped completely.

(d) Procedure for suspending transfer

To suspend transfer:

1. Write 1 to the CLREN bit in the CHCTRL_n register.
2. If the SBE bit in the CHCFG_n register is 0, DMAC stops according to the value of the REQD bit in the CHCFG_n register. If the SBE bit is 1, DMAC is placed in sweep mode.
3. Read the CHSTAT_n register to check whether the TACT bit is 0. If the TACT bit is 0, DMAC has stopped completely. If the TACT bit is 1, continue polling until the bit changes to 0.
4. To perform the next DMA transfer after it is suspended, make sure that the SWRST (software reset) bit in the CHCTRL_n register is turned on before the next DMA transfer starts.

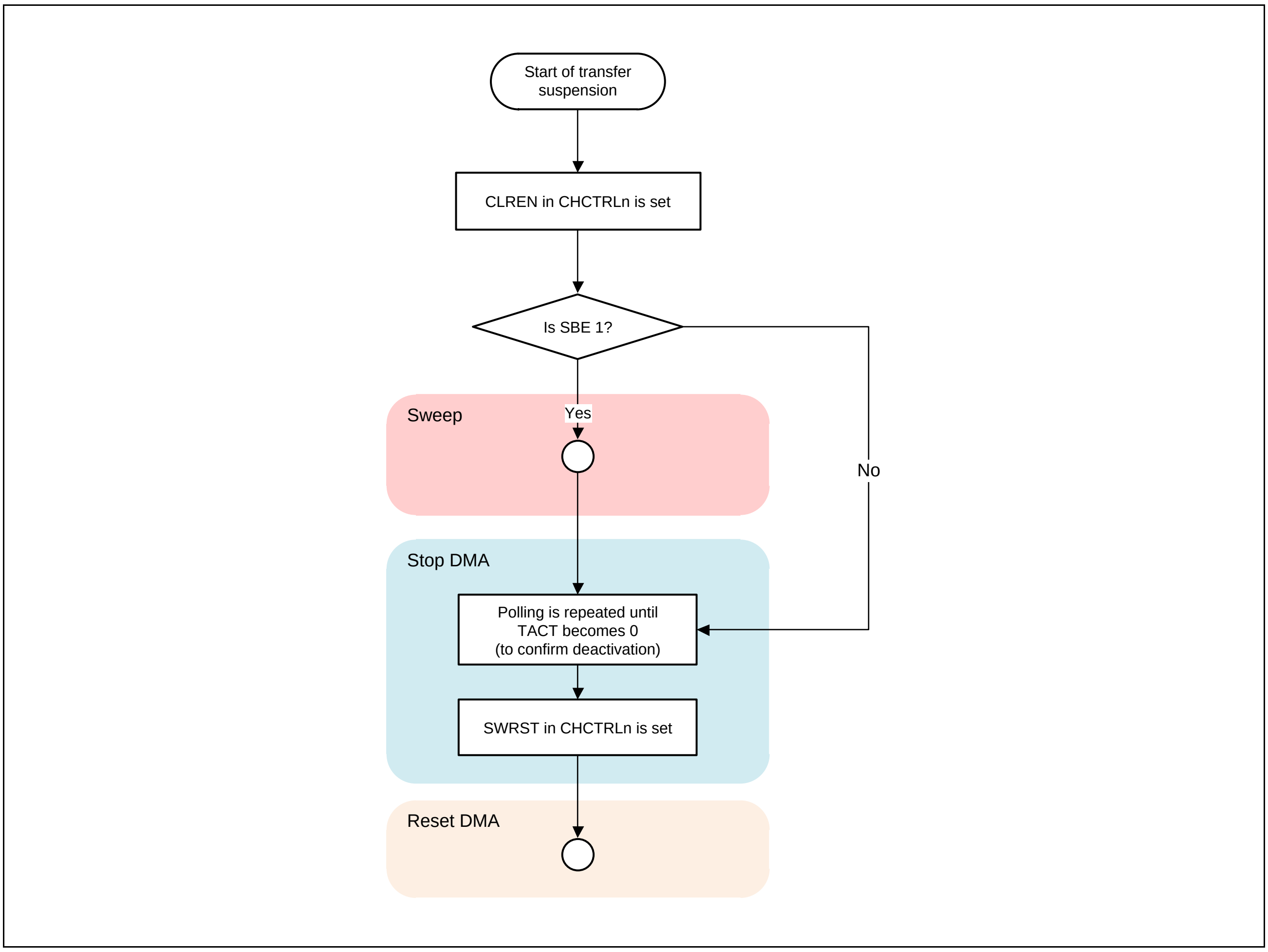


Figure 32B.31 Transfer Suspension Flowchart

32B.9.14 Access Type

32B.9.14.1 DMA master transfer combination list

(1) Read access

The following describes characteristics of the issuance type for DMA read access.

- An access is performed for a beat align space whose size is set by SDS[2:0] in the CHCFG_n register, including the source address indicated in the CRSA_n register. A beat unaligned transfer for the bus is not performed. An excess area is sometimes read depending on CRSA_n or SKIP settings. In this case, the necessary data is imported into the buffer from the read data.
- The size and burst length are determined based on the value set in the SDS[2:0] field.
  If the value set in the SDS[2:0] field is equal to or less than the bus width
    Size: Value set in SDS[2:0]
    Burst type: SINGLE
  If the value set in the SDS[2:0] field is larger than the bus width
    Size: Bus width
    Burst type: Fixed-length burst (burst length = SDS[2:0] value/bus width)

The following indicates the access types for the bus.

Table 32B.53 DMA Read Transfer Combination List

		AHB transfer					
		First transfer			Second transfer		
SDS	Source Address	Address	Data Size	Burst Type	Address	Data Size	Burst Type
0 (8 bits)	-	addr	8	SINGLE			
1 (16 bits)	2 byte align	{addr[31:1], 0b}	16	SINGLE			
	2 byte unalign				{addr[31:1], 0b} + H’2	16	SINGLE
2 (32 bits)	4 byte align	{addr[31:2], 00b}	32	SINGLE			
	4 byte unalign				{addr[31:2], 00b} + H’4	32	SINGLE
4 (128 bits)	16 byte align	{addr[31:4], H’0}	32	INCR4			
	16 byte unalign				{addr[31:4], H’0} + H’10	32	INCR4
5 (256 bits)	32 byte align	{addr[31:5], H’00}	32	INCR8			
	32 byte unalign				{addr[31:5], H’00} + H’20	32	INCR8
6 (512 bits)	64 byte align	{addr[31:6], H’00}	32	INCR16			
	64 byte unalign				{addr[31:6], H’00} + H’40	32	INCR16

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(2) Write access

The following describes characteristics of the issuance type for DMA write access.

- An access is performed from the destination address indicated by the CRDA_n register to the beat align boundary whose size is set by DDS[2:0] in the CHCFG_n register.
- The size and burst length are determined based on the value set in the DDS[2:0] field.
  If the value set in the DDS[2:0] field is equal to or less than the bus width
    Size: Value set in DDS[2:0]
    Burst type: SINGLE
  If the value set in the DDS[2:0] field is greater than the bus width
    Size: Bus width
    Burst type: Fixed-length burst (burst length = DDS[2:0] value/bus width)
- In write access, only the specified space is accessed. In the following cases, the combination of values smaller than the value set in the DDS[2:0] field is used for access.
  - The destination address is beat-unaligned for the value set in the DDS[2:0] field.
  - An access specified in the DDS field will be across the SKIP boundary.
  - The size specified in the DDS[2:0] field is too large for the number of remaining bytes to be transferred.

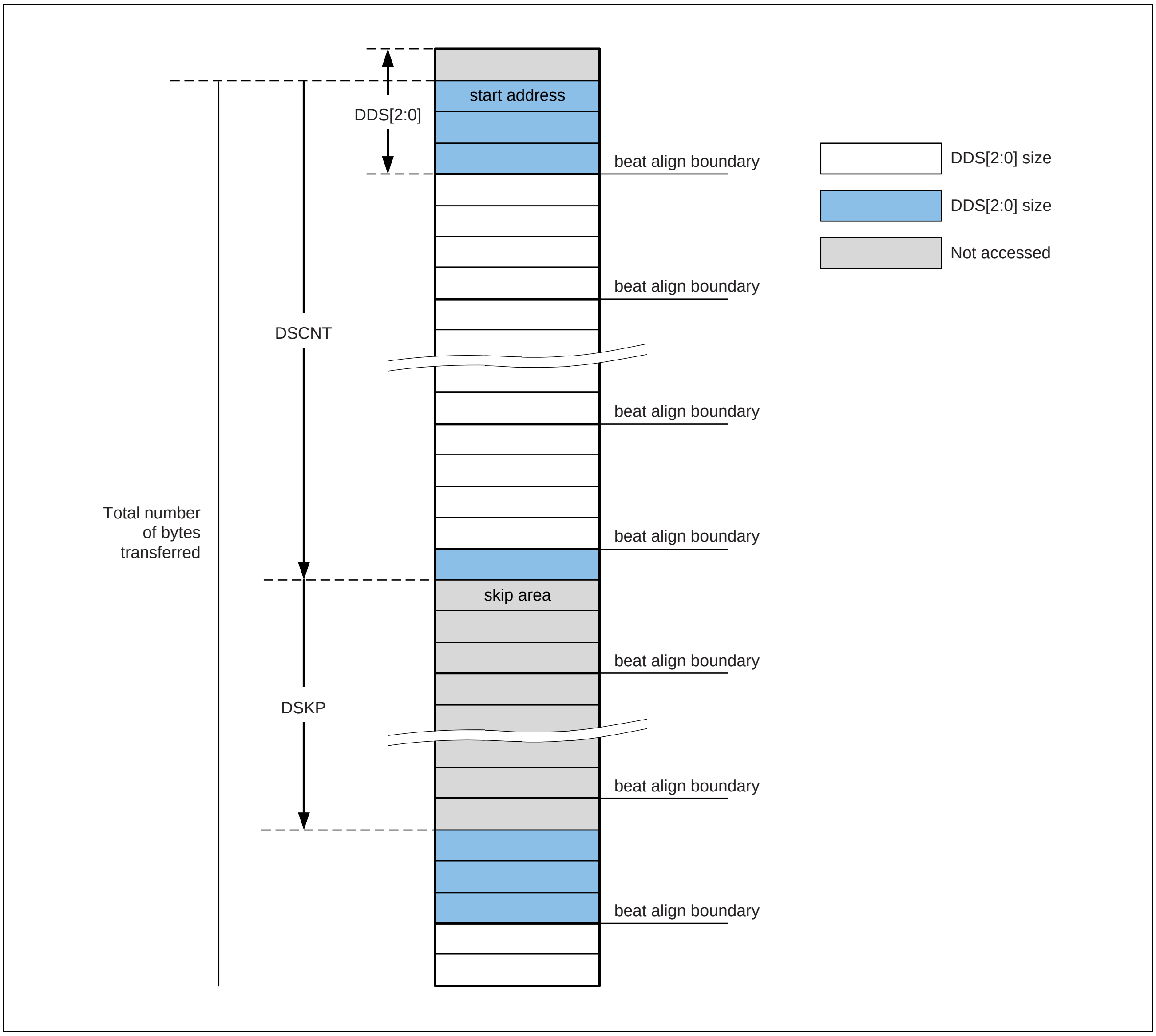


Figure 32B.32 Example of DMA Write Access Space and Access Types

The following shows the access types for the bus in the case of beat-aligned.

Table 32B.54 DMA Write Transfer Combination List

	AHB transfer		
	First transfer		
DDS[2:0]	Address	Data Size	Burst Type
0 (8 bits)	addr	8	SINGLE
1 (16 bits)	{addr[31:1], 0b}	16	SINGLE
2 (32 bits)	{addr[31:2], 00b}	32	SINGLE
4 (128 bits)	{addr[31:4], H'0}	32	INCR4
5 (256 bits)	{addr[31:5], H'00}	32	INCR8
6 (512 bits)	{addr[31:6], H'00}	32	INCR16

32B.9.14.2 DMA master descriptor combination list

(1) Read access

The following describes characteristics of a descriptor access.

- Accesses are performed for 8-word beat align space including the LINK address (the address indicated in the CRLA_n register). A beat unaligned transfer is not performed.
- The size and burst length set by INCR8 are used.
- The descriptor format (DSCFM) of the read header is analyzed, and then the descriptor data is set in an internal register.
- If the descriptor extends over the 8-word boundary, an additional read is performed on the succeeding eight words.

The following indicates the types of access to the bus.

Table 32B.55 Descriptor Read Transfer Combination List

Descriptor Format	Address	AHB Transfer					
		First Transfer			Second Transfer		
	addr[4:0]	Address	Size	Burst	Address	Size	Burst
4 words	H’00	{addr[31:4], 0000b}	32	INCR8			
	H’04						
	H’08						
	H’0C						
	H’10						
	H’14				{addr[31:4], 0000b} + H’20	32	INCR8
	H’18						
	H’1C						
8 words	H’00						
	H’04				{addr[31:4], 0000b} + H’20	32	INCR8
	H’08						
	H’0C						
	H’10						
	H’14						
	H’18						
	H’1C						

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(2) Write access

Use single transfer as the issuance type for writing data back to a descriptor. The following indicates the types of access to the bus.

Table 32B.56 Descriptor Write Transfer Combination List

Type	AHB transfer		
	Address	Size	Burst
Write-back in normal mode	{addr[31:2], 00b} + H’3	8	SINGLE
Write-back in the case of an error	{addr[31:2], 00b} + H’2	8	SINGLE

32B.9.15 Arbitration between DMACs

Arbitration between internal DMACs is performed in round-robin mode.

In round-robin mode, the highest priority is given to the DMAC whose DMAC number is the DMC number being used for transfer + 1. Immediately after a reset, DMAC0 has the highest priority.

Table 32B.57 Priority of a transfer request for DMACs that are performing transfer

Current DMAC	Next DMAC	
	DMAC0	DMAC1
DMAC0	2	1
DMAC1	1	2

Note: Priority: 1 (high), 2 (low)

32B.9.16 Notes

32B.9.16.1 Access

During read access, a beat align area is accessed by one transfer. Therefore, if beat unaligned is set, the beat align area including the specified area is accessed.

For example, if the Source Address is H’0000_1038 and SDS is 5 (256-bit), read starts from area H’0000_1020, not from address H’0000_1038. At this time, if the area from H’0000_1020 to H’0000_1037 contains a register whose value changes by read, the operation might be disrupted.

To prevent problems, use the beat align setting to access a register whose value changes by read access or access to an area adjacent to FIFO.

32B.9.16.2 Level Interrupt bit

This is a DMA interrupt output enable bit. Always set this bit to 1 irrespective of whether peripheral module interrupt USBFDMAmn (m, n = 0, 1) or USBFDMAERRm (m = 0, 1) is used.

To use peripheral module interrupt USBFDMAmn or USBFDMAERRm, set this bit to 1, and then set up the interrupt controller.

33. LCDC

33.1 Overview

This chapter describes the features of the LCDC unit of this LSI.

This unit is LCD controller, it is composed of Frame Compression Processor (FCPVD), Video Signal Processor (VSPD), and Display Unit (DU)

33.1.1 Features

The following is key features of this unit.

- FCPVD
  - Support out-of-order for the whole outstanding transactions
  - Read linear addressing image data
  - Read display list data
  - Write image data
- VSPD
  - Supports various data formats and conversion
    - Supports YCbCr444/422/420, RGB, α RGB, α plane
    - Color space conversion and changes to the number of colors by dithering
    - Color keying
    - Supports combination between pixel alpha and global alpha
    - Supports generating pre multiplied alpha
  - Video processing
    - Blending of two picture layers and raster operations (ROPs)
    - Clipping
    - 1D look up table
    - Vertical flipping in case of output to memory
  - Direct connection to display module
    - Supporting 1920 pixels in horizontal direction
    - Writing back image data which is transferred to Display Unit (DU) to memory
- DU
  - Supporting Display Parallel Interface (DPI) and MIPI LINK Video Interface
  - Display timing master
    - Generating video timings (Front porch, Back porch, Sync active, Active video area)
    - Selecting the polarity of output DCLK, HSYNC, VSYNC, and DE
  - Supporting Progressive (Non-interlace)
  - Not supports Interlace

- Input data format (from VSPD): RGB888, RGB666 (not supports dithering of RGB565)
- Output data format: same as Input data format
- Supporting Full HD (1920 pixels × 1080 lines) for MIPI-DSI Output
- Supporting WXGA (1280 pixels × 800 lines) for Parallel Output

33.1.2 Block Diagram

Figure 33.1 shows a block diagram.

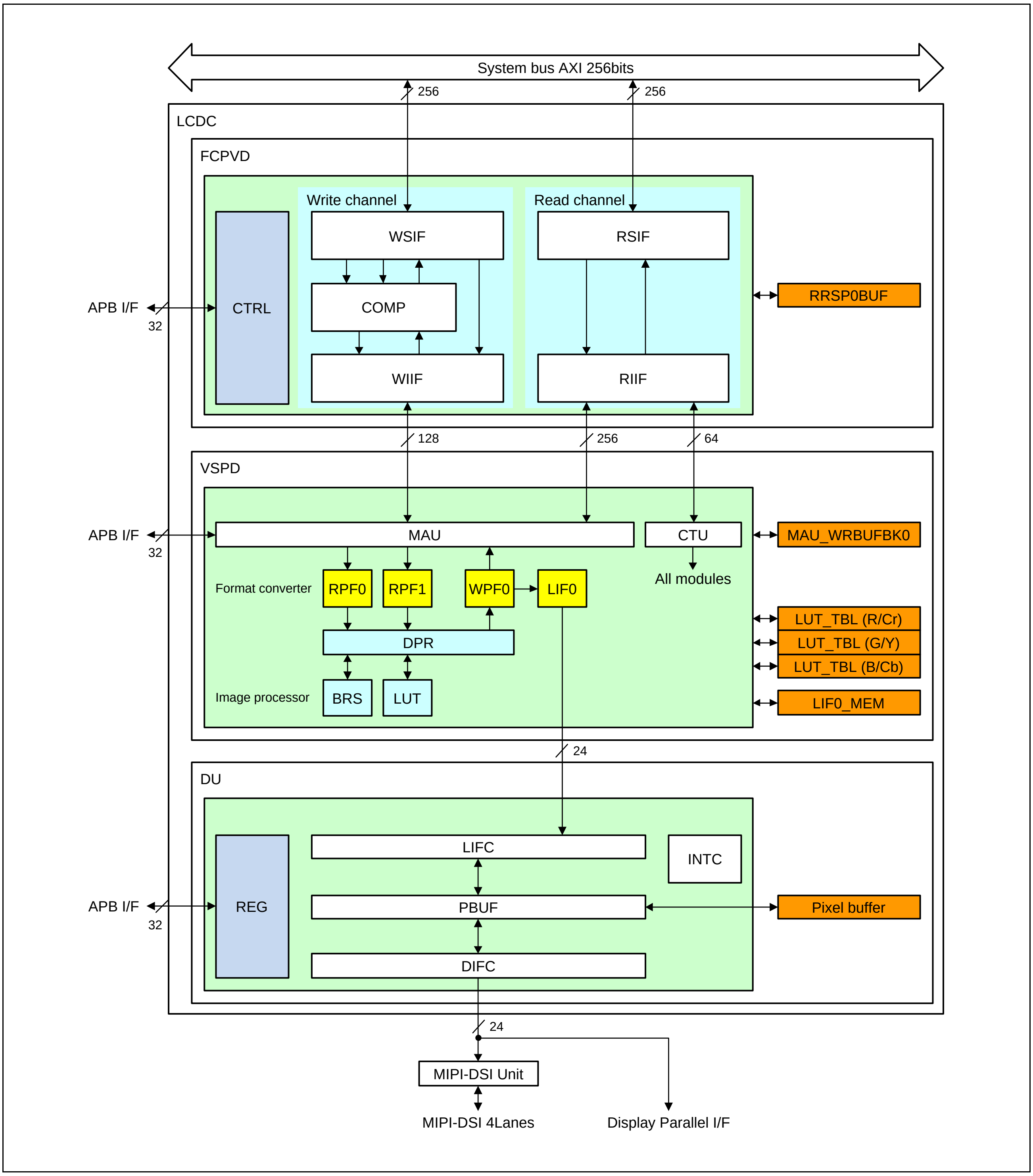


Figure 33.1 Block Diagram

33.1.2.1 FCPVD

FCPVD has the following sub modules.

Table 33.1 Sub modules in FCPVD

Abbreviation	Description
CTRL	FCPVD controller
WIIF	VSPD write channel interface
WSIF	System AXI write channel interface
RIIF	VSPD read channel interface
RSIF	System AXI read channel interface
COMP	Compression module (This is a dummy module)

33.1.2.2 VSPD

VSPD has the following sub modules.

(1) Memory Access Unit (MAU)

The VSPD applies processing to the image data stored in the external memory and writes the resultant data back to the external memory. The data transfer between the external memory and VSPD necessary for this operation is done by the MAU, which works as the bus master, according to the register settings. The MAU executes this data transfer between the external memory and VSPD.

(2) Command Transfer Unit (CTU)

The VSPD can directly read register parameters for image processing by display lists stored in external memory. The CTU module is a bus interface and controls display lists when the CTU reads display lists as a bus master.

(3) Read Pixel Formatter (RPF)

The RPF reads image data from the external memory through the MAU, unpacks data according to the specified format, converts the color space, converts the number of colors, executes color keying, ROP operation, Multiply-alpha and OSD processing, and outputs the resultant data to the DPR. The RPF has an input format unpacking unit, a 1-bit mask generator, a raster operation unit (ROP unit), a color keying unit, a color space converter and multiply-alpha. **Figure 33.2** show the processing flow in the RPF.

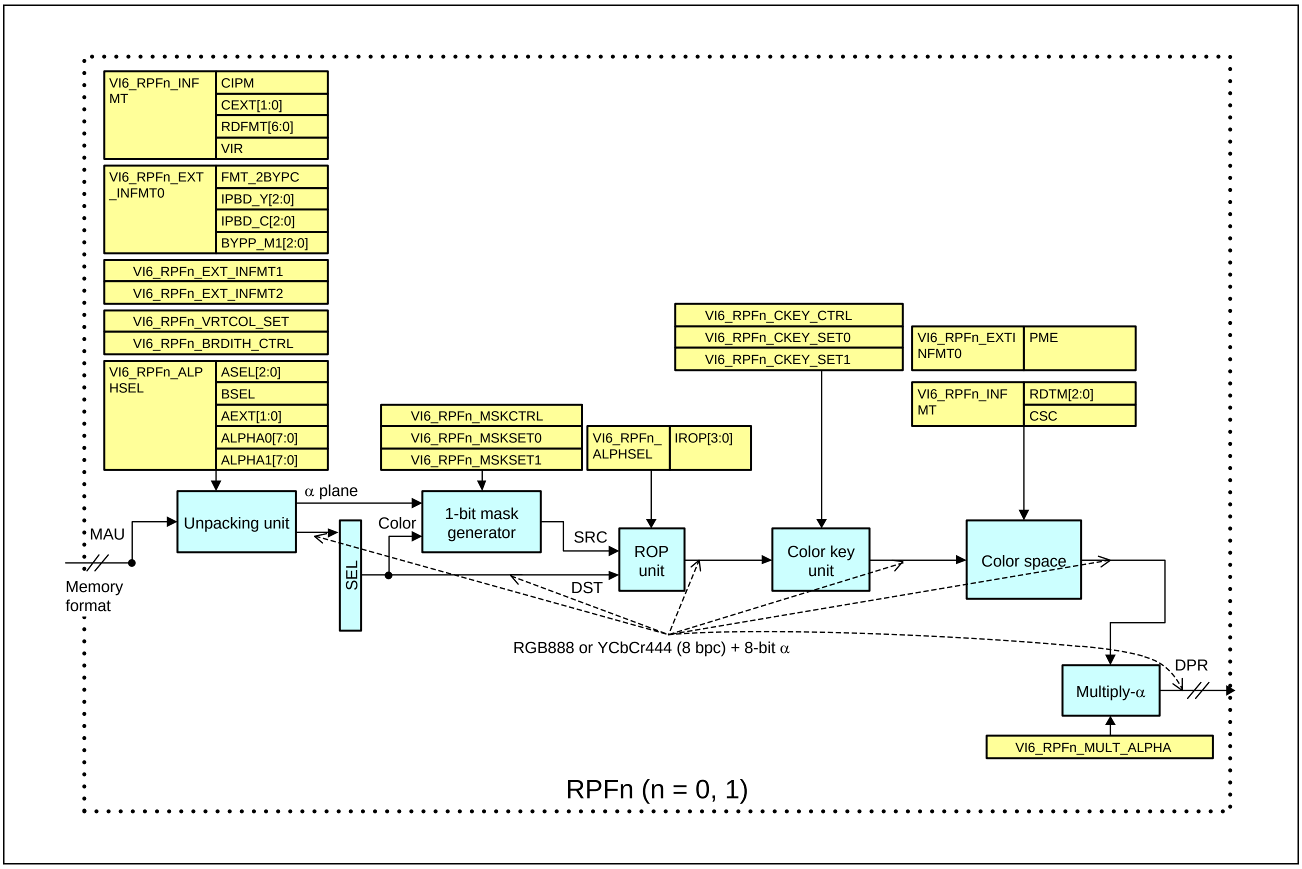


Figure 33.2 RPF Processing Flow

The input format unpacking unit expand the image data input from the MAU into the image format for internal processing (YCbCr444 8bpc or RGB888 8bpc), and the 1-bit mask generator generates a 1-bit image mask from the image data expanded through the unpacking unit. Alternately, a 1-bit image mask can be generated from the alpha plane that is different from the picture plane read through the MAU.

The raster operation unit (ROP unit) executes raster operation between the data from the 1-bit mask generator and the image data expanded from the input format, and the color keying unit applies color replacement and specifies the transparent color for the image data input from the ROP unit. The color space converter converts the color space (RGB-YCbCr) of the image data input from the color keying unit as necessary. The multiply-alpha unit multiplies pixel-alpha by specified alpha or/and multiplies image data by pixel-alpha or/and specified alpha.

The VSPD provides maximum two RPF modules (RPF0 to RPF1).

(4) Data Path Router (DPR)

The DPR controls the data paths among RPFs, function modules, and WPFs. The DPR selects one of the images input from RPFs, outputs it to a function module (BRS or LUT), and selects one of WPFs as the destination where the image data processed in the function module will be output. Before output to the WPF, the output from each function module can be input to another function module, which enables multiple image processing functions to be executed continuously without involving the external memory.

(5) Look Up Table (LUT)

This is a 1D-LUT that converts each of three color components by using a lookup table. The LUT is connected to the DPR and can be used for gamma correction, negative-positive conversion, posterization, and binarization through desired tone curve settings.

The LUT supports local area processing for Y component. The LUT can apply 1D look up table for Y component per area where an image is divided into multiple area specified by registers and command list.

(6) Blend ROP Sub Unit (BRS)

The BRS is a module connected to the DPR, which executes the image blending processing and ROP operation. The BRS has two blend/ROP operation units (blend/ROP unit m, m = A to B), a blend/ROP input switch (SEL) for selecting the input to these operation units, and a divider for normalization (div unit).

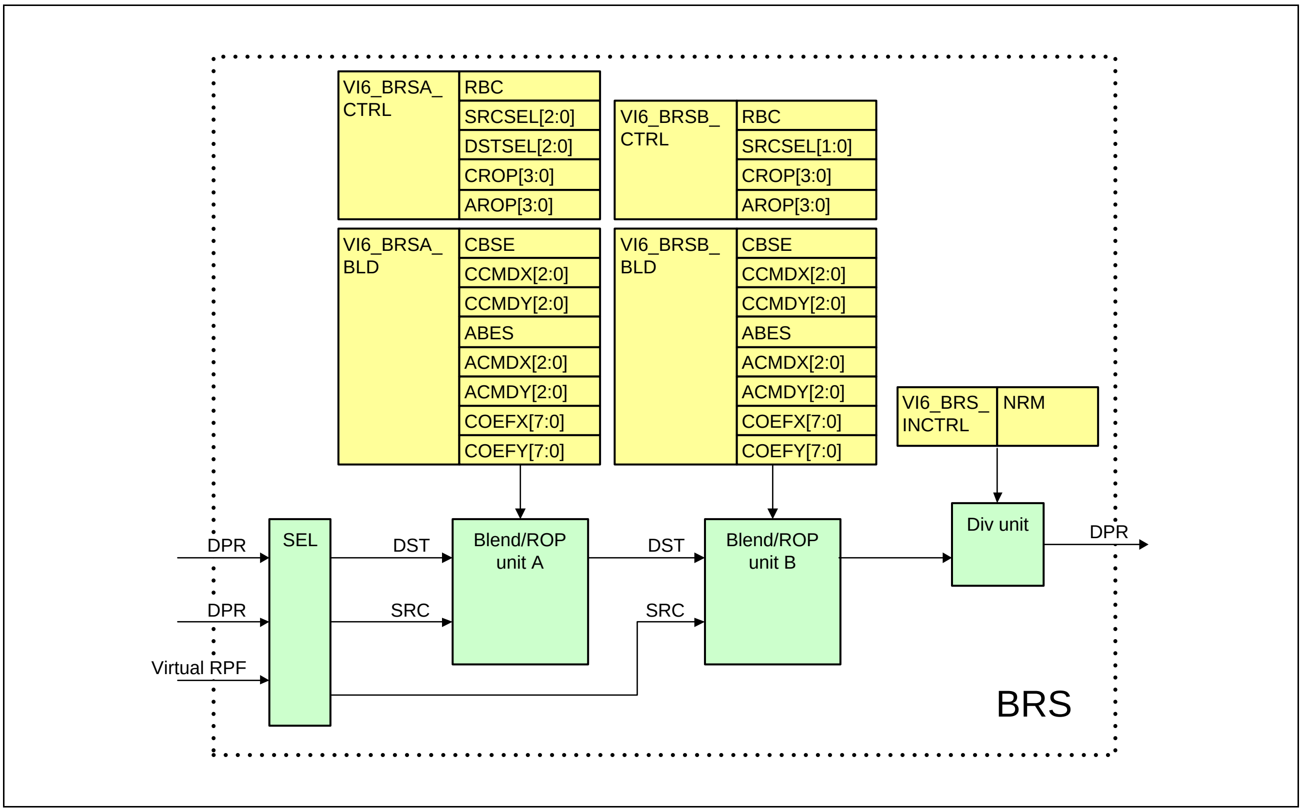


Figure 33.3 BRS Processing Flow

Each of the two blend-ROP operation units (blend/ROP unit m) receives the output from the SEL or blend/ROP Unit m, and executes blending or raster operation (ROP) of images.

The divider for normalization (div unit) divides the pixel value by the α value.

(7) Write Pixel Formatter (WPF)

The WPF is an output module that receives 32bits image data (YCbCr444 or RGB888 + 8-bitα) from the DPR, converts the color space, number of colors, and format of the data, and outputs the results of VSPD image processing to external memory through the MAU. The WPF is mainly configured from a color space converter and an output format converter (the packing unit).

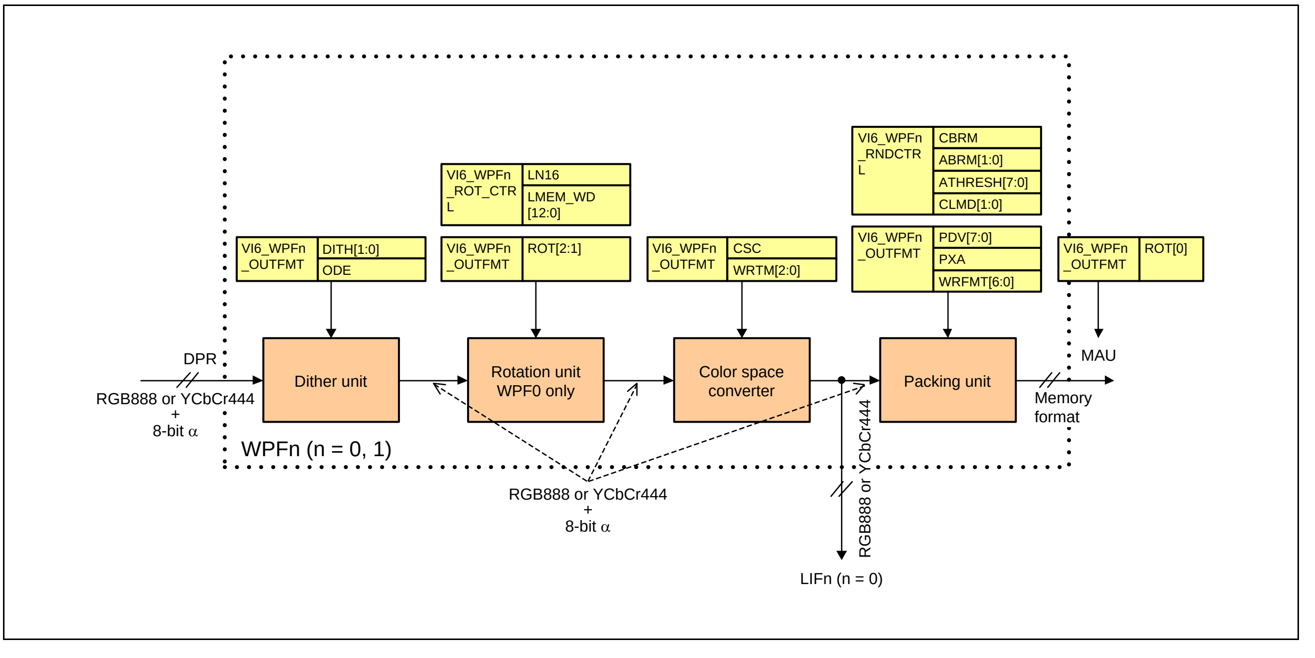


Figure 33.4 WPF Processing Flow

The color space converter converts the color space between RGB and YCbCr, and the packing unit converts the format into the picture plane storing format. The VSPD provides one WPF (WPF0). The WPF has the output to LIF module after the color space conversion for transferring the video data to the display module. Image data to LIF is 24bits (YCbCr444 or RGB888) format, and its' color space is after color space conversion.

(8) Display Unit InterFace (LIF)

The LIF module is used for transferring image data to the display module. The input port of the LIFn (n = 0) module is connected to WPFn (n = 0), and the output port of the LIF module is connected to DU (Display Unit). Data flow in LIF is shown in **Figure 33.30**.

(9) Detail Function

Table 33.2 Detail Function of VSPD (1/5)

● Image Data Transfer Function			
Bus interface	Protocol		AXI 256 bits through FCPVD
	Data alignment	Conversion method	Byte, Word, LW, or LLW data swapping
		Channel	Data alignment can be specified separately for each input/output channel
Image memory	Input	Address setting	1-byte units
	Output	Address setting	1-byte units
		Memory area	Images can be written to the same memory area where the master layer is stored. Note the restrictions shown in **Table 33.39**
Tile transfer mode			Supported by RPF0 to RPF1
YCbCr memory storage format			Interleaved, planar, or semi-planar
● Display list/Extended Display List Transfer Function			
Bus interface	Protocol		AXI 256 bits through FCPVD
	Data alignment	Conversion method	Byte, Word, LW data swapping
			Data alignment can be specified separately for display list and each entry of extended display list
Data memory	Load	Address setting	8-byte units
	Store	—	Not available

Table 33.2 Detail Function of VSPD (2/5)

● Read Pixel Formatter (RPF)			
Number of channels			Two channels (RPF0 to RPF1)
Input color bit-depth		Color	2 to 8bit
		Alpha	1/2/4/8 bit
Operation bit-depth		Color	8bit
		Alpha	8bit
Image format	Input	RGB	RGB888, RGB565, RGB666, αRGB8888, αRGB4444, αRGB1555 α plane (8 bpp, 1 bpp)
		YCbCr	YCbCr4:4:4 (8 bpc) Planar/Semi planar/Interleaved YCbCr4:2:2 (8 bpc) Planar/Semi planar/Interleaved YCbCr4:2:0 (8 bpc) Planar/Semi planar/Interleaved α plane (8 bpp, 1 bpp)
		Maximum size	1920 × 1080 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to **Section 33.5.3**.
		Minimum size	1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to **Section 33.5.3**.
		Size setting unit	YCbCr420: 2-pixel units both horizontally and vertically. YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
Color keying	Color replacement	Compared data	RGB or Y (8 bpc)
		Replaced data	αRGB or αYCbCr (8 bpc for R, G, B, Y, Cb, Cr, α)
		Comparison Mode	Matched color mode
		Input source	RPF0 to RPF1
	Transparent color	Compared data	RGB or Y (8 bpc)
		Replaced data	α 8bpp
		Comparison Mode	Matched color mode, Luma threshold mode
		Input source	RPF0 to RPF1
Raster operation	ROP2 (within input channels)	Operator	16 types (OpenGL2.0 is supported)
		Sources of operation	ROP2 operation between the 1-bpp α plane and RGB/YCbCr data in RPF0 to RPF1. Note that 1-bpp α is converted to αRGB or αYCbCr4:4:4. (8 bpc)
Color space conversion	RGB to YCbCr	Conversion expression	RGB (0, 2^n-1) to BT601 (2^{n-4}, 235*2^{n-8}/240*2^{n-8}) RGB (0, 2^n-1) to BT709 (2^{n-4}, 235*2^{n-8}/240*2^{n-8}) RGB (0, 2^n-1) to BT601 (0, 2^n-1) RGB (2^{n-4}, 235*2^{n-8}) to BT709 (2^{n-4}, 235*2^{n-8}/240*2^{n-8}) (n = 8)
		Target of conversion	RPF0 to RPF1
	YCbCr to RGB	Conversion expression	BT.601 (2^{n-4}, 235*2^{n-8}/240*2^{n-8}) to RGB (0, 2^n-1) BT.709 (2^{n-4}, 235*2^{n-8}/240*2^{n-8}) to RGB (0, 2^n-1) BT.601 (0, 2^n-1) to RGB (0, 2^n-1) BT.709 (2^{n-4}, 235*2^{n-8}/240*2^{n-8}) to RGB (2^{n-4}, 235*2^{n-8}) (n = 8)
		Target of conversion	RPF0 to RPF1

Table 33.2 Detail Function of VSPD (3/5)

● Read Pixel Formatter (RPF)			
Changing number of colors	Bit extension	Extended bits	Input bitdepth (2bpc to 6bpc) to operation bitdepth 8bpc
		Target format	RGB666, RGB565, RGB555, RGB444, or RGB332
		Bit reduction method	Padded with 0. Copied from the most significant bits.
		Target of conversion	RPF0 to RPF1
	YCbCr444 generation	Vertical	CbCr copying
		Horizontal	copying or interpolation.
α bit count conversion	Bit extension	Extended bits	Input bitdepth 1bpc to operation bitdepth 8bpc
		Method	Padded with 0. Copied from the most significant bits.
Multiply-alpha function	Fade-alpha		RPF0 to RPF1 Available for straight pixel and pre-multiplied pixel.
	Generate pre-multiplied alpha		RPF0 to RPF1
Virtual display	RPF0 to RPF1	Color format	αRGB8888 or αYCbCr4:4:4 single-color
		Display size	Same as the size of the input channel
	Virtual RPF	Color format	αRGB8888 or αYCbCr4:4:4 single-color
		Display size	Maximum: 1920 × 1080 pixels Minimum: 4 × 4 pixels

Table 33.2 Detail Function of VSPD (4/5)

• Write Pixel Formatter (WPF)			
Number of channels			One channel (WPF0)
Image format	Output	RGB	RGB332, RGB444, RGB565, RGB666, RGB888, αRGB8666, αRGB8888, αRGB4444, αRGB1555
		YCbCr	YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0
		Maximum size	1920 × 1080 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to **Section 33.5.4**.
		Minimum size	1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to **Section 33.5.4**.
		Size setting unit	YCbCr420: 2-pixel units both horizontally and vertically. YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
Color space conversion	RGB to YCbCr	Conversion expression	RGB (0, 255) to BT.601 (16, 235/240) RGB (0, 255) to BT.709 (16, 235/240) RGB (0, 255) to BT.601 (0, 255) RGB (16, 235) to BT.709 (16, 235/240)
		Target of conversion	WPF0
	YCbCr to RGB	Conversion expression	BT.601 (16, 235/240) to RGB (0, 255) BT.709 (16, 235/240) to RGB (0, 255) BT.601 (0, 255) to RGB (0, 255) BT.709 (16, 235/240) to RGB (16, 235)
		Target of conversion	WPF0
Changing number of colors	Output	Reducing RGB color depth	Dithering, lower-order bit truncation, or rounding
		YCbCr422/420	CbCr skipping or CbCr vertical skipping and horizontal skipping
α bit count conversion	Output	Bit reduction	Truncation, rounding, or comparison with threshold (for 1 bpp)

Table 33.2 Detail Function of VSPD (5/5)

● Image Compositing			
α blending	Input α value selection	RGB	Pixel α, fixed α value, α plane, or 1-bit α converted from the color specified for pixels
		YCbCr	α plane, fixed α value, or 1-bit α converted from the color specified for pixels
	α blending expression	Layer A: Upper layer Layer B: Lower layer	xA + yB, xA - yB Coefficients x and y should be selected from the following. Fixed α value, (α for layer A), (1 - α for layer A), (α for layer B), (1 - α for layer B)
	Output α value selection	RGB	Fixed α value, x (α for layer A) + y (α for layer B), x (α for layer A) – y (α for layer B) Coefficients x and y should be selected from the following. Fixed α value, (α for layer A), (1 - α for layer A), (α for layer B), (1 - α for layer B)
	Blending layers	Number of layers	Two layers selected from RPF0 to RPF1 and video processing function output, and virtual RPF; three layers in total
		Order of layers	The order of three layers selected from RPF0 to RPF1, virtual RPF, and video processing function output can be changed as desired.
	α plane	Format	8 bpp or 1 bpp (α value can be specified through register)
		Input source	RPF0 to RPF1
	Fixed α value	Format	8 bpp
		Input source	RPF0 to RPF1, virtual RPF, or video processing function output
Raster operation	ROP2 (between input channels)	Operator	16 types (OpenGL2.0 is supported)
		Sources of operation	RPF0 to RPF1, virtual RPF, and video processing function output
		Operation control	RGB/YCbCr and α are operated separately.
	ROP3 (between input channels)	Operator	256-type ROP3 is available by combining ROP2 operations
		Sources of operation	RPF0 to RPF1, virtual RPF, and video processing function output
		Operation control	RGB/YCbCr and α are operated separately.
● Color Adjustment Function			
1D-LUT	LUT configuration	Independent R/Y, G/Cb, and B/Cr. 256 entries each	
Direct connection to Display module			
Display I/F	Destination		DU
	Direct connection		Can transfer data without going through external memory to DU
Line padding	Padding cycles		1 to 32 cycles
	Padding pattern		Arbitrarily setting up to 32 cycles

33.1.2.3 DU

DU has the following sub modules.

(1) REG

The REG controls registers for DU. The DU has APB4 interface, its transaction is transfer with no wait states.

(2) LIFC

The LIFC is the interface block from/to VSPD. The LIFC receives image data from VSPD, sends to PBUF in a pixel unit. The LIFC continues to receive the image data from VSPD unless FIFO will be full.

(3) PBUF

The PBUF is the asynchronous FIFO which convert clock domain of image data from System Bus Clock to Video Clock. The PBUF watches over FIFO underflow.

(4) DIFC

The DIFC is the display timing master. The DIFC controls the polarity and the timing of the control signals (DISP_CLK, DISP_HSYNC, DISP_VSYNC, and DISP_DE). Controllable timings are the followings.

- Hfront: Horizontal front porch (pixels)
- Hback: Horizontal back porch (pixels)
- Hsync: Horizontal sync active (pixels)
- Hactive: Horizontal active area (pixels)
- Vfront: Vertical front porch (lines)
- Vback: Vertical back porch (lines)
- Vsync: Vertical sync active (lines)
- Vactive: Vertical active area (lines)

(5) INTC

The INTC generates the interrupt when PBUF will detect FIFO underflow. The interrupt is level signal.

33.1.3 External Pins

Table 33.3 shows the pin configuration.

Table 33.3 LCDC Display Parallel Interface (DPI) External Pins

Pin Name	Input/Output	Function
DISP_CLK	Output	Display Parallel Interface pixel clock
DISP_HSYNC	Output	Display Parallel Interface Horizontal sync pulse
DISP_VSYNC	Output	Display Parallel Interface Vertical sync pulse
DISP_DE	Output	Display Parallel Interface data enable
DISP_DATA0	Output	Display Parallel Interface pixel data output Red0
DISP_DATA1	Output	Display Parallel Interface pixel data output Red1
DISP_DATA2	Output	Display Parallel Interface pixel data output Red2
DISP_DATA3	Output	Display Parallel Interface pixel data output Red3
DISP_DATA4	Output	Display Parallel Interface pixel data output Red4
DISP_DATA5	Output	Display Parallel Interface pixel data output Red5
DISP_DATA6	Output	Display Parallel Interface pixel data output Red6
DISP_DATA7	Output	Display Parallel Interface pixel data output Red7
DISP_DATA8	Output	Display Parallel Interface pixel data output Green0
DISP_DATA9	Output	Display Parallel Interface pixel data output Green1
DISP_DATA10	Output	Display Parallel Interface pixel data output Green2
DISP_DATA11	Output	Display Parallel Interface pixel data output Green3
DISP_DATA12	Output	Display Parallel Interface pixel data output Green4
DISP_DATA13	Output	Display Parallel Interface pixel data output Green5
DISP_DATA14	Output	Display Parallel Interface pixel data output Green6
DISP_DATA15	Output	Display Parallel Interface pixel data output Green7
DISP_DATA16	Output	Display Parallel Interface pixel data output Blue0
DISP_DATA17	Output	Display Parallel Interface pixel data output Blue1
DISP_DATA18	Output	Display Parallel Interface pixel data output Blue2
DISP_DATA19	Output	Display Parallel Interface pixel data output Blue3
DISP_DATA20	Output	Display Parallel Interface pixel data output Blue4
DISP_DATA21	Output	Display Parallel Interface pixel data output Blue5
DISP_DATA22	Output	Display Parallel Interface pixel data output Blue6
DISP_DATA23	Output	Display Parallel Interface pixel data output Blue7

33.1.4 Clock and Interrupt signal

The following clocks are supplied to the LCDC.

Table 33.4 Clock List

Clock Name	Clock Symbol	Description	Frequency (MHz)
LCDC_CLK_A	M0ϕ	AXI clock. This clock should be a synchronous clock of LCDC_CLK_P	200
LCDC_CLK_P	ZTϕ	APB clock. This clock should be a synchronous clock of LCDC_CLK_A	100
LCDC_CLK_D	M3ϕ	Video clock. The duty of this clock is 50% when the display parallel interface	up to 148.5

The LCDC transmits the following interrupt signals to the interrupt controller.

Table 33.5 Interrupt List

Signal Name	Interrupt Condition	Interrupt Source Register
VSPD_INT	Frame End	VI6_WPF0_IRQ_STA.FRE
	Display List Frame End	VI6_WPF0_IRQ_STA.DFE
	DU Connection UnderRun Error	VI6_WPF0_IRQ_STA.UND
DU_INT	RBUF FIFO Underflow	DU_MSR0.ST_PB_RUF

Note: Refer to **Section 33.3.2.4(13)**. WPFn Interrupt Status Registers and **Section 33.3.3.1**. DU Module Control Register 0 for details.

33.2 Register Configuration

33.2.1 FCPVD Register Configuration

The FCPVD contains the registers which can directly be read and written to by the host CPU. The host can access any long word (32 bits) in the registers. Therefore, read-modify-write is needed to change partial bits in 32bits register.

Never access the registers when the FCPVD is in the module standby mode (with the operation clock placed in hold status).

Base Address: H’0_1088_0000 (Cortex-A55 Address Space)
H’4088_0000 (Cortex-M33 Address Space Non-Secure)
H’5088_0000 (Cortex-M33 Address Space Secure)

Table 33.6 FCPVD Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
FCPVD version control register	FCP_VCR	R	H’0000 0109	H’0000	32

33.2.2 VCPD Register Configuration

Base Address: H’0_1087_0000 (Cortex-A55 Address Space)
H’4087_0000 (Cortex-M33 Address Space Non-Secure)
H’5087_0000 (Cortex-M33 Address Space Secure)

Lowercase “n” in a register name or a offset address indicates an integer and the range of value n is defined when necessary. For RPFn, and WPFn, when the range of value n is not defined, RPFn (n = 0, 1), WPFn (n = 0) are assumed.

Table 33.7 VCPD Register Configuration (1/3)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
VSPD Start Registers 0	VI6_CMD0	R/W*1	H’0000 0000	H’0000	32
Clock Control Register 0	VI6_CLK_CTRL0	R/W	H’0000 0000	H’0010	32
Clock Control Register 1	VI6_CLK_CTRL1	R/W	H’0000 0000	H’0014	32
Dynamic Clock Stop Control Register	VI6_CLK_DCSWT	R/W	H’0000 0000	H’0018	32
Dynamic Clock Stop Disable Register 0	VI6_CLK_DCSM0	R/W	H’0000 0000	H'001c	32
Dynamic Clock Stop Disable Register 1	VI6_CLK_DCSM1	R/W	H’0000 0000	H'0020	32
Software Reset Register	VI6_SRESET	R/W	H’0000 0000	H'0028	32
Module Reset Enable Register 0	VI6_MRESET_ENB0	R/W	H’0000 0000	H'002c	32
Module Reset Enable Register 1	VI6_MRESET_ENB1	R/W	H’0000 0000	H'0030	32
Module Reset Issuing Register	VI6_MRESET	R/W	H’0000 0000	H'0034	32
Operating Status Register	VI6_STATUS	R	H’0000 0000	H'0038	32
WPF0 Interrupt Enable Register	VI6_WPF0_IRQ_ENB	R/W	H’0000 0000	H'0048	32
WPF0 Interrupt Status Register	VI6_WPF0_IRQ_STA	R/W	H’0000 0000	H'004c	32
Display-0 Interrupt Enable Register	VI6_DISP0_IRQ_ENB	R/W	H’0000 0000	H'0078	32
Display-0 Interrupt Status Register	VI6_DISP0_IRQ_STA	R/W	H’0000 0000	H'007c	32
Display List Control Register	VI6_DL_CTRL	R/W	H’0000 0000	H'0100	32
Display List-0 Header Address Register	VI6_DL_HDR_ADDR0	R/W	H’0000 0000	H'0104	32
Display List-0 Data Swapping Register	VI6_DL_SWAP0	R/W	H’0000 0000	H'0114	32
Extended Display List-0 Control Register	VI6_DL_EXT_CTRL0	R/W	H’0000 0000	H'011c	32
Display List Body Size Register 0	VI6_DL_BODY_SIZE0	R/W	H’0000 0000	H'0120	32
Display List-0 Header Reference Address Register	VI6_DL_HDR_REF_ADDR0	R	H’0000 0000	H'0130	32
Display List-0 Wake Up Counter Register	VI6_DL_WUPCNT0	R/W	H’0000 0000	H'0158	32
RPFn Basic Read Size Registers	VI6_RPFn_SRC_BSIZE	R/W	H’0000 0000	H'0300 + H'0100*n	32
RPFn Extended Read Size Registers	VI6_RPFn_SRC_ESIZE	R/W	H’0000 0000	H'0304 + H'0100*n	32

Table 33.7 VCPD Register Configuration (2/3)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
RPFn Input Format Registers	VI6_RPFn_INFMT	R/W	H’0000 0000	H'0308 + H'0100*n	32
RPFn Data Swapping Registers	VI6_RPFn_DSWAP	R/W	H’0000 0000	H'030c + H'0100*n	32
RPFn Display Location Registers	VI6_RPFn_LOC	R/W	H’0000 0000	H'0310 + H'0100*n	32
RPFn α Plane Selection Control Registers	VI6_RPFn_ALPH_SEL	R/W	H’0000 0000	H'0314 + H'0100*n	32
RPFn Virtual Plane Color Information Registers	VI6_RPFn_VRTCOL_SET	R/W	H’0000 0000	H'0318 + H'0100*n	32
RPFn Mask Control Registers	VI6_RPFn_MSKCTRL	R/W	H’0000 0000	H'031c + H'0100*n	32
RPFn IROP-SRC Input Value Registers 0	VI6_RPFn_MSKSET0	R/W	H’0000 0000	H'0320 + H'0100*n	32
RPFn IROP-SRC Input Value Registers 1	VI6_RPFn_MSKSET1	R/W	H’0000 0000	H'0324 + H'0100*n	32
RPFn Color Keying Control Registers	VI6_RPFn_CKEY_CTRL	R/W	H’0000 0000	H'0328 + H'0100*n	32
RPFn Color Keying Color Setting Registers 0	VI6_RPFn_CKEY_SET0	R/W	H’0000 0000	H'032c + H'0100*n	32
RPFn Color Keying Color Setting Registers 1	VI6_RPFn_CKEY_SET1	R/W	H’0000 0000	H'0330 + H'0100*n	32
RPFn Source Picture Memory Stride Setting Registers	VI6_RPFn_SRCM_PSTRIDE	R/W	H’0000 0000	H'0334 + H'0100*n	32
RPFn Source α Memory Stride Setting Registers	VI6_RPFn_SRCM_ASTRIDE	R/W	H’0000 0000	H'0338 + H'0100*n	32
RPFn Source Y/RGB Address Registers	VI6_RPFn_SRCM_ADDR_Y	R/W	H’0000 0000	H'033c + H'0100*n	32
RPFn Source Chroma Address Registers 0	VI6_RPFn_SRCM_ADDR_C0	R/W	H’0000 0000	H'0340 + H'0100*n	32
RPFn Source Chroma Address Registers 1	VI6_RPFn_SRCM_ADDR_C1	R/W	H’0000 0000	H'0344 + H'0100*n	32
RPFn Source α Address Registers	VI6_RPFn_SRCM_ADDR_AI	R/W	H’0000 0000	H'0348 + H'0100*n	32
RPFn Bus Access Control Registers	VI6_RPFn_BAC	R/W	H’0000 0000	H'0350 + H'0100*n	32
RPFn Multiple Alpha Control Registers	VI6_RPFn_MULT_ALPH	R/W	H’0000 0000	H'036C+ H'0100*n	32
WPFn-Source-RPF Registers	VI6_WPFn_SRCRPF	R/W	H’0000 0000	H'1000 + H'0100*n	32
WPFn Horizontal Input Size Clipping Registers	VI6_WPFn_HSZCLIP	R/W	H’0000 0000	H'1004 + H'0100*n	32
WPFn Vertical Input Size Clipping Registers	VI6_WPFn_VSZCLIP	R/W	H’0000 0000	H'1008 + H'0100*n	32
WPFn Output Format Registers	VI6_WPFn_OUTFMT	R/W	H’0000 0000	H'100c + H'0100*n	32
WPFn Data Swapping Registers	VI6_WPFn_DSWAP	R/W	H’0000 0000	H'1010 + H'0100*n	32
WPFn Rounding Control Registers	VI6_WPFn_RNDCTRL	R/W	H’0000 0000	H'1014 + H'0100*n	32
WPFn Destination Y Plane Memory Stride Registers	VI6_WPFn_DSTM_STRIDE_Y	R/W	H’0000 0000	H'101c + H'0100*n	32
WPFn Destination C Plane Memory Stride Registers	VI6_WPFn_DSTM_STRIDE_C	R/W	H’0000 0000	H'1020 + H'0100*n	32

Table 33.7 VCPD Register Configuration (3/3)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
WPFn Destination Y/RGB Address Registers	VI6_WPFn_DSTM_ADDR_Y	R/W	H’0000 0000	H'1024 + H'0100*n	32
WPFn Destination Chroma Address Registers 0	VI6_WPFn_DSTM_ADDR_C0	R/W	H’0000 0000	H'1028 + H'0100*n	32
WPFn Destination Chroma Address Registers 1	VI6_WPFn_DSTM_ADDR_C1	R/W	H’0000 0000	H'102c + H'0100*n	32
WPF0 LIF Write Back Control Registers	VI6_WPF0_WRBCK_CTRL	R/W	H’0000 0000	H'1034	32
RPF0 Routing Register	VI6_DPR_RPF0_ROUTE	R/W	H’0000 0000	H'2000	32
RPF1 Routing Register	VI6_DPR_RPF1_ROUTE	R/W	H’0000 0000	H'2004	32
WPF0 Timing Control Register	VI6_DPR_WPF0_FPORCH	R/W	H’0000 0000	H'2014	32
LUT Routing Register	VI6_DPR_LUT_ROUTE	R/W	H’0000 0000	H'203c	32
BRS Routing Register	VI6_DPR_BRS_ROUTE	R/W	H’0000 0000	H'2050	32
LUT Control Register	VI6_LUT_CTRL	R/W	H’0000 0000	H'2800	32
BRS Input Control Register	VI6_BRS_INCTRL	R/W	H’0000 0000	H'3900	32
Size Register of BRS Input Virtual RPF	VI6_BRS_VIRRPF_SIZE	R/W	H’0000 0000	H'3904	32
Display Location Register of BRS Input Virtual RPF	VI6_BRS_VIRRPF_LOC	R/W	H’0000 0000	H'3908	32
Color Information Register of BRS Input Virtual RPF	VI6_BRS_VIRRPF_COL	R/W	H’0000 0000	H'390c	32
BRS Control Registers A	VI6_BRSA_CTRL	R/W	H’0000 0000	H'3910	32
BRS Blend Control Registers A	VI6_BRSA_BLD	R/W	H’0000 0000	H'3914	32
BRS Control Registers B	VI6_BRSB_CTRL	R/W	H’0000 0000	H'3918	32
BRS Blend Control Registers B	VI6_BRSB_BLD	R/W	H’0000 0000	H'391c	32
LIF0 Control Registers	VI6_LIF0_CTRL	R/W	H’0000 0000	H'3b00	32
LIF0 Clock Stop Buffer Control Register	VI6_LIF0_CSBTH	R/W	H’0000 0000	H'3b04	32
LIF0 Buffer Attribute Register	VI6_LIF0_LBA	R/W	H’0000 0000	H'3b0c	32
LIF0 Padding Line Cycle Register	VI6_LIF0_PADLN_CYC	R/W	H’0000 0000	H'3b30	32
LIF0 Padding Line Pattern Register	VI6_LIF0_PADLN_PT	R/W	H’0000 0000	H'3b34	32
LIF0 Padding Line Value Register	VI6_LIF0_PADLN_VAL	R/W	H’0000 0000	H'3b38	32
LIF0 Padding Line Size Register	VI6_LIF0_PADLN_SIZE	R/W	H’0000 0000	H'3b3C	32
LUT table	VI6_LUT_TBL_0 To VI6_LUT_TBL_255	R/W	Undefined	H'7000 To H'73fc	32

Note 1. Read only bits or write only bits or read/write bits are mixed in the registers.

33.2.3 DU Register Configuration

The DU contains the registers which can directly be read and written to by the host CPU. The host can access any long word (32 bits), word (16 bits), and byte (8 bits) in the registers. Therefore, read-modify-write is needed to change partial bits not more than 8 bits in 32 bits register.

Never access the registers when the DU is in the module standby mode (with the operation clock placed in hold status).

Base Address: H’0_1089_0000 (Cortex-A55 Address Space)
H’4089_0000 (Cortex-M33 Address Space Non-Secure)
H’5089_0000 (Cortex-M33 Address Space Secure)

Table 33.8 DU Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DU Module Control Register 0	DU_MCR0	R/W	H'00000000	H'00	32/16/8
DU Module Status Register 0	DU_MSR0	R	H'00540000	H'04	32/16/8
DU Module Status Register 1	DU_MSR1	R	H'00000000	H'08	32/16/8
DU Interrupt Mask Register 0	DU_IMR0	R/W	H'00000000	H'0C	32/16/8
DU Display I/F Timing Register 0	DU_DITR0	R/W	H'00030300	H'10	32/16/8
DU Display I/F Timing Register 1	DU_DITR1	R/W	H'00000000	H'14	32/16/8
DU Display I/F Timing Register 2	DU_DITR2	R/W	H'00000000	H'18	32/16/8
DU Display I/F Timing Register 3	DU_DITR3	R/W	H'00000000	H'1C	32/16/8
DU Display I/F Timing Register 4	DU_DITR4	R/W	H'00000000	H'20	32/16/8
DU Module Control Register 1	DU_MCR1	R/W	H’00000000	H’40	32/16/8
DU PBUF Control Register 0	DU_PBCR0	R/W	H'00000000	H'4C	32/16/8
DU PBUF Control Register 1	DU_PBCR1	R/W	H’00000000	H’50	32/16/8
DU PBUF Control Register 2	DU_PBCR2	R/W	H’00000000	H’54	32/16/8

33.3 Register Descriptions

33.3.1 FCPVD Registers

33.3.1.1 FCPVD version Control Register (FCP_VCR)

FCP_VCR is the version control register of the FCPVD. The value read is fixed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATEGORY								REVISION							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 8	CATEGORY	H'01	R	FCP use case category.
7 to 0	REVISION	H'09	R	Version of LSI product H'09: FCP revision is this LSI.

33.3.2 VCPD Registers

33.3.2.1 Notational Conventions for Registers and Bit Fields

This section uses the following notational conventions for the VSPD registers and bit fields.

1. The names of registers and bits are written in uppercase.
2. A bit or bit field in a register is indicated as [register name.bit name]. For example, the STRCMD bit in the VI6_CMD0 register is indicated as VI6_CMD0.STRCMD.
3. Lowercase “n” in a register name or a bit name indicates an integer and the range of value n is defined when necessary. For RPFn, and WPFn, when the range of value n is not defined, RPFn (n = 0, 1), WPFn (n = 0) are assumed.
4. In each subsection for register description in **Section 33.3.2**, when only a bit name is written without showing its register name, the bit is in the register described in that subsection.
5. A wildcard (*) indicates any characters in a name and represents all registers or bits that match the specified first part of a name. For example, when there are two registers VI6_RPF_SRC_BSIZE and VI6_RPF_SRC_ESIZE, VI6_RPF_SRC_* indicates both registers.

33.3.2.2 Register Classification

The VSPD registers are arranged in the following order; the general control registers to display list control registers control operation of the entire VSPD, and the other registers control each image processing and specify parameters for the processing. The functions of the registers are described in this order starting from **Section 33.3.2.4**.

1. General control registers (VI6_CMDn (n = 0, 1), VI6_SRESET, VI6_STATUS, VI6_WPFn_IRQ_* (n = 0))
2. Display list control registers (VI6_DL_*)
3. RPF control registers (VI6_RPFn_* (n = 0, 1))
4. WPF control registers (VI6_WPFn_* (n = 0))
5. DPR control registers (VI6_DPR_*)
6. LUT control register (VI6_LUT_CTRL)
7. BRS control registers (VI6_BRS_*)
8. LIFn control registers (VI6_LIFn_* (n = 0))

The VSPD has two RPF channels and the register configuration is the same for all of RPFn (n = 0, 1). However, some bit fields have restrictions in certain RPFs. These restrictions are included in the description of the corresponding bit fields and registers. Likewise, the register configuration is the same for all WPFn (n = 0), but some bit fields have restrictions in certain WPFs; the restrictions are included in the description of the corresponding bit fields and registers.

For each register address, refer to **Table 33.7**.

33.3.2.3 Restrictions on Access to Registers and Lookup Tables

The VSPD has control registers and lookup tables. All VSPD registers are writable and readable by only 32 bits unit. To write partial bits in each register, read-modify-write is needed. When accessing the addresses where these registers and lookup tables are allocated, the following restrictions should be satisfied. If any restriction is violated, the VSPD will not operate correctly.

1. For the read-only bits and reserved bits in all VSPD registers, writing 1 is prohibited unless otherwise specified.
2. Addresses undefined in **Table 33.7** are reserved areas and write access is prohibited in these areas.
3. For all registers and lookup tables, except VI6_CMDn, VI6_SRESET, VI6_*IRQ* and two plane registers such as VI6_DL_HDR_ADDRn, VI6_DL_BODY_SIZE0, modifying register values during operation of the module is prohibited. Modify registers while the corresponding module is stopped. For the operating status of the target module, refer to **Section 33.4.4**.
4. There are three types about General control registers (**Section 33.3.2.4**) and Display List Control Registers (**Section 33.3.2.5**) as below.
    - Controlling WPF0 (ex. VI6_CMD0, VI6_SRESET.SRST0)
    - Common setting of WPF0 (ex. VI6_DL_CTRL.AR_WAIT [15:0])

Table 33.9 Correspondence between Modules and Register Names

Module Name	Register Name
RPFn (n = 0, 1)	VI6_RPFn_*
WPFn (n = 0)	VI6_WPFn_*
DPR	VI6_DPR_*
LUT	VI6_LUT_CTRL
BRS	VI6_BRS_*
LIFn (n = 0)	VI6_LIFn_*

33.3.2.4 General Control Registers

(1) VSPD Start Registers n (VI6_CMDn: n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	UPDHDR	—	—	—	STRCMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	UPDHDR	0	R/W	Reserved state of updating Display List Header (DLH) address. UPDHDR can be negated by software reset (VI6_SRESET.SRSTn). [Write] 0: NOP 1: Don't set 1 to this bit [Read] 0: Not reserved to update DLH address. 1: Reserved to update DLH address. Set 0 to this bit when STRCMD bit is set to 1.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STRCMD	0	R/W	Start reservation of WPF VI6_CMDn.STRCMD controls WPFn. Writing 1 to this bit starts WPFn in VSPD. Set this bit for activation only after all register settings in each output channel have been completed. If WPFn is idle, WPFn starts right after this bit is set to 1. If WPFn is active, writing 1 to this bit reserves starting WPFn operation. Wait to set 1 to VI6_CMDn.STRCMD until this bit is read as 0. VI6_CMDn.STRCMD can be negated by software reset (VI6_SRESET.SRSTn). [Write] 0: Start reservation of WPFn is canceled. 1: Start reservation of WPFn is set. [Read] 0: Starting VSPD is not reserved. 1: Starting VSPD is reserved.

The basic concept of image processing operation started by activating the VSPD is shown in **Figure 33.5**. The actual data input/output is executed by the MAU, which is the bus interface module, as described in **Section 33.1.2**, but conceptually the RPF works as the data entry point to the VSPD and the WPF works as the data exit point. To process images through the VSPD, the RPF (entrance) and WPF (exit) should be connected and a data path from the RPF to the WPF should be formed.

To connect RPFn to WPFn, specify RPFn as the source RPF for WPFn in VI6_WPFn_SRCRPF, which is a register for WPFn (refer to **Section 33.3.2.7(1)**). This setting determines that RPFn will be started when WPFn is started through VI6_CMDn.

A data path to execute desired image processing should then be formed between the RPF (entrance) and WPF (exit). To form a data path, connect the necessary function modules in the VSPD between RPF and WPF. This function is provided by the DPR; specify the information for each module connection in data path routing registers VI6_DPR_*_ROUTE (refer to **Section 33.3.2.8**).

After a data path is formed (RPFn → WPFn) as described above, starting output module WPFn in the VSPD through VI6_CMDn starts all function modules connected to WPFn and the desired image processing is executed. According to this design concept, starting a WPF module means starting the VSPD.

There are two types of data path configuration in the VSPD; one is “a single input module to a single output module” as shown in **Figure 33.5 (A)**, and the other is “multiple input modules to a single output module” as shown in **Figure 33.5 (B)**.

Figure 33.5 (A) shows an example of a configuration where modules with single input and single output are implemented through the DPR.

Figure 33.5 (B) shows another configuration example where the module with multiple input and single output is implemented through the DPR.

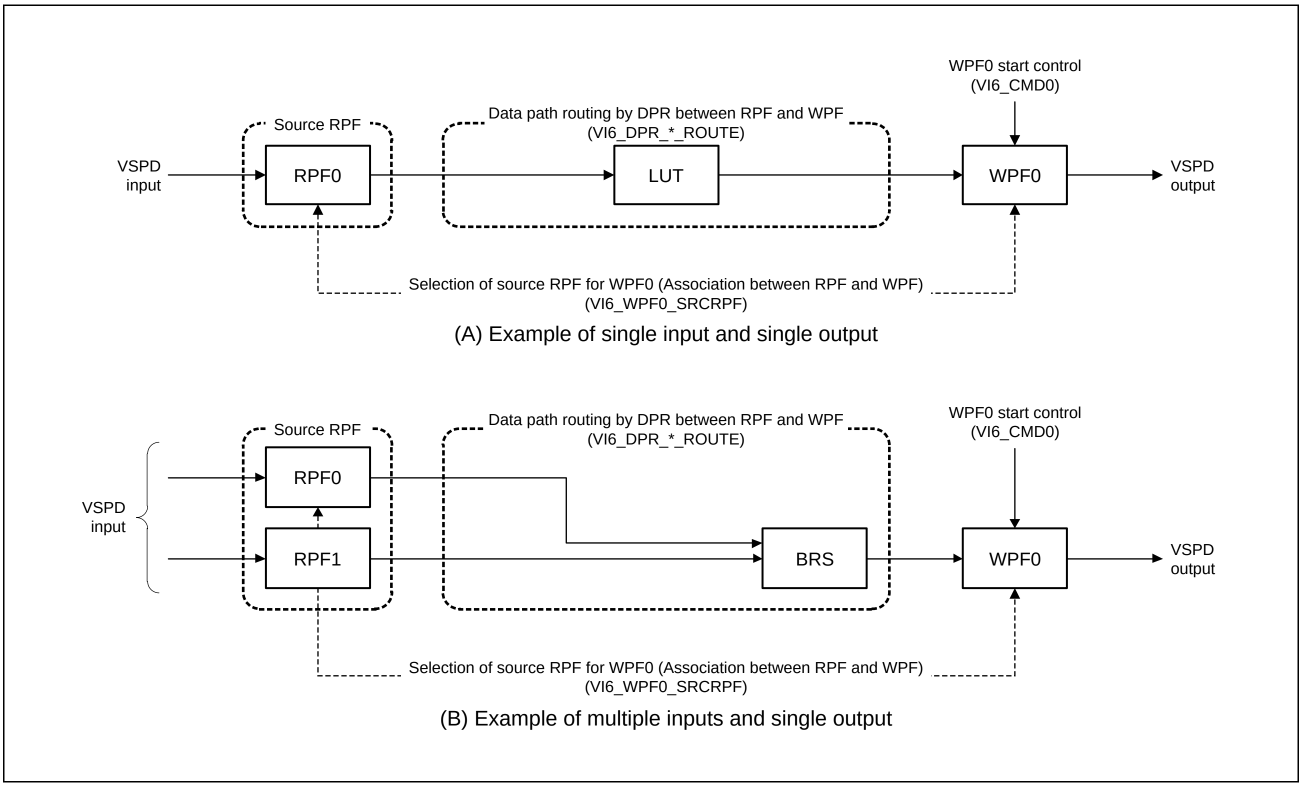


Figure 33.5 Basic Concept of VSPD Startup

(2) Clock Control Register 0 (VI6_CLK_CTRL0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	GCS0	—	—	—	—	—	—	—	—	—	—	—	GCS1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	GCS2				—	—	—	GCS3				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	GCS0	0	R/W	Clock Control Setting0
27 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	GCS1	0	R/W	Clock Control Setting1
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	GCS2	0	R/W	Clock Control Setting2
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	GCS3	0	R/W	Clock Control Setting3

VSPD can stop its operating clock for reducing power consumption.

To enable clock stop function, set following registers:

- VI6_CLK_CTRL0 = H’0000 0000
- VI6_CLK_CTRL1 = H’0000 0000
- VI6_CLK_DCSWT = H’0000 0808
- VI6_CLK_DCSM0 = H’0000 0000
- VI6_CLK_DCSM1 = H’0000 0000

To disable clock stop function, set following registers:

- VI6_CLK_CTRL0 = H’1001 0F1F
- VI6_CLK_CTRL1 = H’FF10 FFFF
- VI6_CLK_DCSWT = H’0033 0808
- VI6_CLK_DCSM0 = H’1FFF 0F1F
- VI6_CLK_DCSM1 = H’FF10 FFFF

When the clock gating is enabled, the supply of the clock signal is stopped for the module which is not operated. If the bit field of this register is changed from 0 to 1, the supply of the clock signal for the module is immediately started. If the bit field of the register is changed from 1 to 0, the clock supply is immediately stopped. Therefore, please do not change this register field from 1 to 0 in operation because the VSPD may be stalled.

(3) Clock Control Register 1 (VI6_CLK_CTRL1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GCS4								—	—	—	GCS6	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GCS5															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GCS4	0	R/W	Clock Control Setting4
23 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	GCS6	0	R/W	Clock Control Setting6
19 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	GCS5	0	R/W	Clock Control Setting5

See **Section 33.3.2.4(2)** for detail.

(4) Dynamic Clock Stop Control Register (VI6_CLK_DCSWT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	DCC0		—	—	DCC1	DCC2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSTPW[7:0]								CSTRW[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21, 20	DCC0	0	R/W	Dynamic Clock Control Setting0
19, 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	DCC1	0	R/W	Dynamic Clock Control Setting1
16	DCC2	0	R/W	Dynamic Clock Control Setting2
15 to 8	CSTPW[7:0]	0	R/W	Dynamic Clock Stop Control 1 Always specify 8.
7 to 0	CSTRW[7:0]	0	R/W	Dynamic Clock Stop Control 2 Always specify 8.

See **Section 33.3.2.4(2)** for detail.

(5) Dynamic Clock Stop Disable Register 0 (VI6_CLK_DCSM0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DCD0												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DCD1				—	—	—	DCD2				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	DCD0	0	R/W	Dynamic Clock Stop Disable Setting0
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	DCD1	0	R/W	Dynamic Clock Stop Disable Setting1
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	DCD2	0	R/W	Dynamic Clock Stop Disable Setting2

See **Section 33.3.2.4(2)** for detail.

(6) Dynamic Clock Stop Disable Register 1 (VI6_CLK_DCSM1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCD3								—	—	—	DCD5	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCD4															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DCD3	0	R/W	Dynamic Clock Stop Disable Setting3
23 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	DCD5	0	R/W	Dynamic Clock Stop Disable Setting5
19 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	DCD4	0	R/W	Dynamic Clock Stop Disable Setting4

See **Section 33.3.2.4(2)** for detail.

(7) Software Reset Register (VI6_SRESET)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SRST0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SRST0	0	R/W	WPFn Software Reset (SRSTn, n = 0) Writing 1 to this bit aborts the current processing in WPFn (the partially-completed image undergoing processing is output). The period until this software reset processing is completed depends on the bus state. When this reset processing is completed, the VI6_WPFn_IRQ_STA.FRE interrupt source bit is set to 1; when the FRE interrupt is enabled, the FRE end interrupt is output to notify the end of the reset processing. This bit is always read as 0. 0: NOP 1: WPFn software reset*[1]

Note 1. Applying a software reset to each WPF has the following restrictions.

1. A software reset can be applied to only one of WPF0 through single write access to VI6_SRESET.
2. After a software reset is issued, no more software reset can be issued to another WPF until the issued software reset processing is completed
3. The end of software reset processing is notified through the FRE bit in VI6_WPFn_IRQ_STA, but the software reset issued while WPF is stopped is ignored as NOP. As it takes a while until the reset is actually issued after the reset bit is set, the VSPD may complete operation before the reset is actually issued. In this case, no interrupt is output for the software reset that is issued after the VSPD completes operation.
4. If a software reset is issued during downloading of a display list, the downloading processing is not aborted. After the end of downloading that is in progress when a software reset is issued, a frame end interrupt is output.

(8) Module Reset Enable Register 0 (VI6_MRESET_ENB0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MRSTE0		—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MRSTE1				—	—	—	MRSTE2				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29, 28	MRSTE0	0	R/W	Module Reset Enable 0
27 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	MRSTE1	0	R/W	Module Reset Enable 1
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	MRSTE2	0	R/W	Module Reset Enable 2

This register is for purpose of h/w debugging.

(9) Module Reset Enable Register 1 (VI6_MRESET_ENB1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MRSTE3								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRSTE4															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MRSTE3	0	R/W	Module Reset Enable 3
23 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	MRSTE4	0	R/W	Module Reset Enable 4

This register is for purpose of h/w debugging.

(10) Module Reset Issuing Register (VI6_MRESET)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	MRST	0	R/W	Module Reset Assertion

This register is for purpose of h/w debugging.

(11) Operating Status Register (VI6_STATUS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	FLDST0	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SYS0_ACT	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	FLDST0	0	R	Field status of previous frame of WPF0. 0: Previous finished frame is TOP field. 1: Previous finished frame is BOT field. This bit can be referred in case of AUTO-FLD or AUTO-DISP. This bit is changed at the timing of VI6_WPF0_IRQ_STA.FRE.
27 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	SYS0_ACT	0	R	WPFn Operating Status (SYSn_ACT, n=0) Each bit indicates the operating or stopped state of control channel n (WPFn). 0: WPFn is stopped. 1: WPFn is operating.
7 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

(12) WPFn Interrupt Enable Registers (VI6_WPFn_IRQ_ENB: n = 0)

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt Disabled

1: Interrupt Enabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UNDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DFEE	FREE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UNDE	0	R/W	Interrupt Enable for WPFn (n = 0) Underrun in case of DU connection.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DFEE	0	R/W	Interrupt Enable for WPFn (n = 0) Display List Frame End
0	FREE	0	R/W	Interrupt Enable for WPFn (n = 0) Frame End

Each bit in VI6_WPFn_IRQ_STA is set to 1 when the corresponding interrupt source is generated. VI6_WPFn_IRQ_ENB specifies whether to output an interrupt signal for the generated source. When an interrupt is disabled in this register, no interrupt signal is generated even when the corresponding bit in VI6_WPFn_IRQ_STA is set to 1. When an interrupt is enabled in this register, an interrupt signal is output when the corresponding bit in VI6_WPFn_IRQ_STA is set to 1.

(13) WPFn Interrupt Status Registers (VI6_WPFn_IRQ_STA: n = 0)

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt

1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0

1: Hold the interrupt status value

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DFE	FRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	UND	0	R/W	Interrupt Status and Clear for WPFn (n = 0) Underrun in case of DU connection. This interrupt source bit is set to 1 when data underrun occurs in case of DU connection. Timing that this bit is set to 1 is at end of 1 frame.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DFE	0	R/W	Interrupt Status and Clear for WPFn (n = 0) Display List Frame End This interrupt source bit is set to 1 when VSPD completes one-frame processing while the current frame in enable value stored in the display list header is 1 (refer to **Section 33.4.4**). When display lists are not used, this bit is not used. In this case, clear VI6_WPFn_IRQ_ENB.DFEE to 0 to mask the interrupt generation by this interrupt source. This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0 to this bit, the oldest interrupt status is cleared. And the status bit becomes 0 after all interrupt statuses are cleared.

Bit	Bit Name	Initial Value	R/W	Description
0	FRE	0	R/W	Interrupt Status and Clear for WPFn (n = 0) Frame End This interrupt source bit is set to 1 when VSPD completes one-frame processing. This bit is also set to 1 when one-frame processing using a display list is completed. The interrupt status is set to 1 by any of the following conditions. a) Processing one frame is finished normally b) Software reset is issued during VSPD is processing. c) One frame's data from DU is displayed while VSPD couldn't transfer one frame data to DU. (in case of Linked with DU) This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0 to this bit, the oldest interrupt status is cleared. And the status bit becomes 0 after all interrupt statuses are cleared.

VI6_WPFn_IRQ_STA indicates the state of the interrupt sources generated in the VSPD. Whether to output a VSPD interrupt when an interrupt source is generated and the corresponding bit is set to 1 is determined by the corresponding bit setting in VI6_WPFn_IRQ_ENB. While an interrupt is disabled in VI6_WPFn_IRQ_ENB, the VSPD does not output an interrupt signal even when an interrupt source is generated, but the source flag in this register is set to 1.

Note that the interrupt source bits in this register cannot be cleared by write access using a display list.

(14) Display-n Interrupt Enable Register (VI6_DISPn_IRQ_ENB: n = 0)

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt Disabled

1: Interrupt Enabled

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DSTE	—	—	MAEE	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	DSTE	0	R/W	Interrupt Enable for Display Start
7 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	MAEE	0	R/W	Interrupt Enable for Display Read Data End
4 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

(15) Display-n Interrupt Status Register (VI6_DISPn_IRQ_STA: n = 0)

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt

1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0

1: Hold the interrupt status value

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DST	—	—	MAE	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	DST	0	R/W	Interrupt Enable for Display Start
7, 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	MAE	0	R/W	Interrupt Enable for Display Read Data End
4 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

33.3.2.5 Display List Control Registers

(1) Display List Control Register (VI6_DL_CTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AR_WAIT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DC2	—	—	—	DC1	—	—	—	DLE1	RLM0	CFM0	NH0	DLE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	AR_WAIT [15:0]	All 0	R/W	Display List Control Setting Always specify 256.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	DC2	0	R/W	This bit doesn't affect anything to VSPD (NOP)
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	DC1	0	R/W	This bit doesn't affect anything to VSPD (NOP)
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	DLE1	0	R/W	This bit doesn't affect anything to VSPD (NOP)
3	RLM0	0	R/W	Loading two plane Registers Mode for WPF0. 0: Reserved status (VI6_CMD0.UPDHDR) is accepted by next frame auto start. Two plane registers are downloaded by next_frame_start also. 1: Reserved status (VI6_CMD0.UPDHDR) is not accepted by next frame auto start. pNextHeader in previous frame's DLH is used for loading DisplayList. Two plane registers are not downloaded by next_frame_start also.
2	CFM0	0	R/W	Continuous Frame Mode for Header-less Display List for WPF0 This bit determines whether the next frame is automatically started or not. When the updated flag of the display list, VI6_DL_BODY_SIZE0.UPD0, is not updated, the display list of the next frame is not transferred and the same register values are used for the next frame. When the value of VI6_DL_BODY_SIZE0.UPD0 is updated, the new display list is transferred. 0: Stopped at the end of every frame 1: The next frame is automatically started

Bit	Bit Name	Initial Value	R/W	Description
1	NH0	0	R/W	Header-less Display List Mode This bit is used for specifying the header-less display list mode. In case of header-less mode, the number of the display lists is 1. The address of the display body is set in VI6_DL_HDR_ADDR0 register, and the body size is set in VI6_DL_BODY_SIZE0 register. When this bit is changed, make sure that VSPD is stopped. And also make sure the following before starting VSPD. – Header Address (VI6_DL_HDR_ADDR0) – Body Size (VI6_DL_BODY_SIZE0) in case of header-less mode 0: Use Display List Header (Normal DL Mode) 1: Don’t use Display List Header (Header-less Mode) *Note 1.* Only WPF0 supports header-less display list. WPF1 work as the normal display list mode even if the WPF0 is set to header-less display list mode. *Note 2.* When DLE0 bit is 0, set 0 to NH0 bit.
0	DLE0	0	R/W	Display List Enable/Disable for WPF0 Enables or disables the WPF0 display list function. When the display list function is enabled through this bit, all WPF processing channels work in display list mode. When using display lists, note the restrictions in **Section 33.3.2.3**. 0: The display list function is disabled 1: The display list function is enabled

(2) Display List-n Header Address Register (VI6_DL_HDR_ADDRn: n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DL_HDR_PTR-n[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL_HDR_PTR-n[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DL_HDR_PTR-n[31:0]	All 0	R/W	Display List-n Header Address These bits specify the address of the display list header to be read for display list-n in 16-byte units (the lower-order four bits are read-only). When WPFn is first started in display list mode, the display list header is loaded from the address specified in this register. After loading of the header is completed, the register value of the display list address is updated to the next header address stored in the loaded header to prepare for loading of the next display list header. After that, this header address updating is repeated. A value from H’0000 0000 to H’FFFF FFF0 can be specified.

(3) Display List-n Data Swapping Register (VI6_DL_SWAPn: n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IND	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	LWS	WDS	BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	IND	0	R/W	Enabling independent swap setting per WPF. This bit is available only for VI6_DL_SWAP1. This bit is reserved for VI6_DL_SWAP0. 0: Display list swap for WPF1 is specified by LWS, WDS and BTS in VI6_DL_SWAP0. 1: Display list swap for WPF1 is specified by LWS, WDS and BTS in VI6_DL_SWAP1
30 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	LWS	0	R/W	Display List Data Swapping in long word Units The effect of this bit setting is defined in **Table 33.10**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled
1	WDS	0	R/W	Display List Data Swapping in Word Units The effect of this bit setting is defined in **Table 33.10**. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled
0	BTS	0	R/W	Display List Data Swapping in Byte Units The effect of this bit setting is defined in **Table 33.10**. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled

Table 33.10 shows the data order before and after swapping according to the long word, word, and byte swapping settings.

When data order in memory for each format is the same as **Table 33.35**, set 111b to {LWS, WDS, BTS}. If data order is different from the definition, change data order within 8byte unit by these bits as shown in **Table 33.10**.

Table 33.10 Changing data order according to display list swap register

Data order in memory									*_LWS	*_WDS	*_BTS
Byte address	8n+0	8n+1	8n+2	8n+3	8n+4	8n+5	8n+6	8n+7			
Data	0	1	2	3	4	5	6	7	1	1	1
	1	0	3	2	5	4	7	6	1	1	0
	2	3	0	1	6	7	4	5	1	0	1
	3	2	1	0	7	6	5	4	1	0	0
	4	5	6	7	0	1	2	3	0	1	1
	5	4	7	6	1	0	3	2	0	1	0
	6	7	4	5	2	3	0	1	0	0	1
	7	6	5	4	3	2	1	0	0	0	0

Data order defined in **Table 33.35**								
Byte address	8n+0	8n+1	8n+2	8n+3	8n+4	8n+5	8n+6	8n+7
Data	0	1	2	3	4	5	6	7

(4) Extended Display List-n Control Register (VI6_DL_EXT_CTRLn: n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NWE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	POLINT						—	—	DLPRI	EXPRI	—	—	—	EXT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	NWE	0	R/W	No Wait for Polling When this bit is set to 1, the polling condition for extended display lists is always assumed to be true.
15, 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 8	POLINT	All 0	R/W	Extended Display List Command Control Always specify 2.
7, 6	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5	DLPRI	0	R/W	Display List Control 0 Always specify 1.
4	EXPRI	0	R/W	Display List Control 1 Always specify 0.
3 to 1	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	EXT	0	R/W	Extended Display List for WPFn Enables or disables the extended display list function. When extended display lists are used, the display list header size is 96 bytes; when they are not used, the header size is 80 bytes. 0: No extended display lists are used 1: Extended display lists are used [Note] When using extended display lists, be sure to also use normal display list mode (VI6_DL_CTRL.DLEn); executing only extended display lists is not possible. When the header-less display list mode is activated, this bit should be set to 0. The extended display list cannot be used with the header-less display list mode. Extended Display List is available only for VSPD/LIF0 or VSPDL/LIFn (n = 0, 1) to realized AUTO-FLD or AUTO-DISP.

(5) Display List Body Size Register n (VI6_DL_BODY_SIZEn: n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	UPD0	—	—	—	—	—	—	—	BS0[16:0]
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BS0[16:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	UPD0	0	R/W	Update Flag This bit controls the download of the display list at the next downloading timing in case that the header-less display list mode is used. Set 1 to this bit when display list need to be downloaded for next frame. This bit is automatically cleared to 0 after VI6_DL_HDR_ADDRn and VI6_DL_BODY_SIZEn.BS0 (n = 0) is downloaded in H/W side. When this bit is set to 1, the value of VI6_DL_HDR_ADDRn and VI6_DL_BODY_SIZEn.BS0 (n = 0) should not be changed. 0: Updating display list for the next frame is not reserved 1: Updating display list for the next frame is reserved
23 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16 to 0	BS0[16:0]	0	R/W	Header-less Display List Body Size (WPFn, n = 0) These bits are used for specifying the body size of the display list in case of header-less display list mode. The unit of the size is byte. The value should be set in multiples of 8.

See **Section 33.4.8.2** for detail.

(6) Display List-n Header Reference Address Register (VI6_DL_HDR_REF_ADDRn: n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DL_HDR_REF_PTR-n [31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL_HDR_REF_PTR-n [31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DL_HDR_REF_PTR-n [31:0]	All 0	R	Display List-n reference Header Address Following value is read out from the VI6_DL_HDR_REF_ADDRn for each period. (1) When H/W is reading display list from external memory, header address of the display list referred by VSPD -H/W is read out. (2) When H/W is not reading display list from external memory, the value of VI6_DL_HDR_ADDRn is read out.

(7) Display List-0 Wake Up Counter Register (VI6_DL_WUPCNT0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

33.3.2.6 RPF Control Registers

(1) RPFn Basic Read Size Registers (VI6_RPFn_SRC_BSIZE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	BHSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BVSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	BHSIZE[12:0]	All 0	R/W	Horizontal Size of RPF Basic Read Area These bits specify the horizontal size of the basic area to be read from the external RAM by the RPFn. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 1920 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPFn_SRC_ESIZE.EHSIZE).
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	BVSIZE[12:0]	All 0	R/W	Vertical Size of RPF Basic Read Area These bits specify the vertical size of the basic area to be read from the external RAM by the RPFn. When the input format is YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 1080 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPFn_SRC_ESIZE.EVSIZE).

Figure 33.6 shows the relationship between the basic read size and extended read size. The RPF reads data from the source memory area specified by the basic read size. The RPF repeats reading the basic read area in the horizontal and vertical directions up to the extended read size and sends the read data to the processing modules in the VSPD.

For basic read size reading, the reading start address, called the RPFn source image storing address, should be specified in VI6_RPFn_SRCM_ADDR_*. In the memory area where the basic read area image is stored, the distance (number of bytes) between addresses for lines n and n + 1 of two-dimensional image data, called the memory stride, should be specified in VI6_RPFn_SRCM_PSTRIDE.

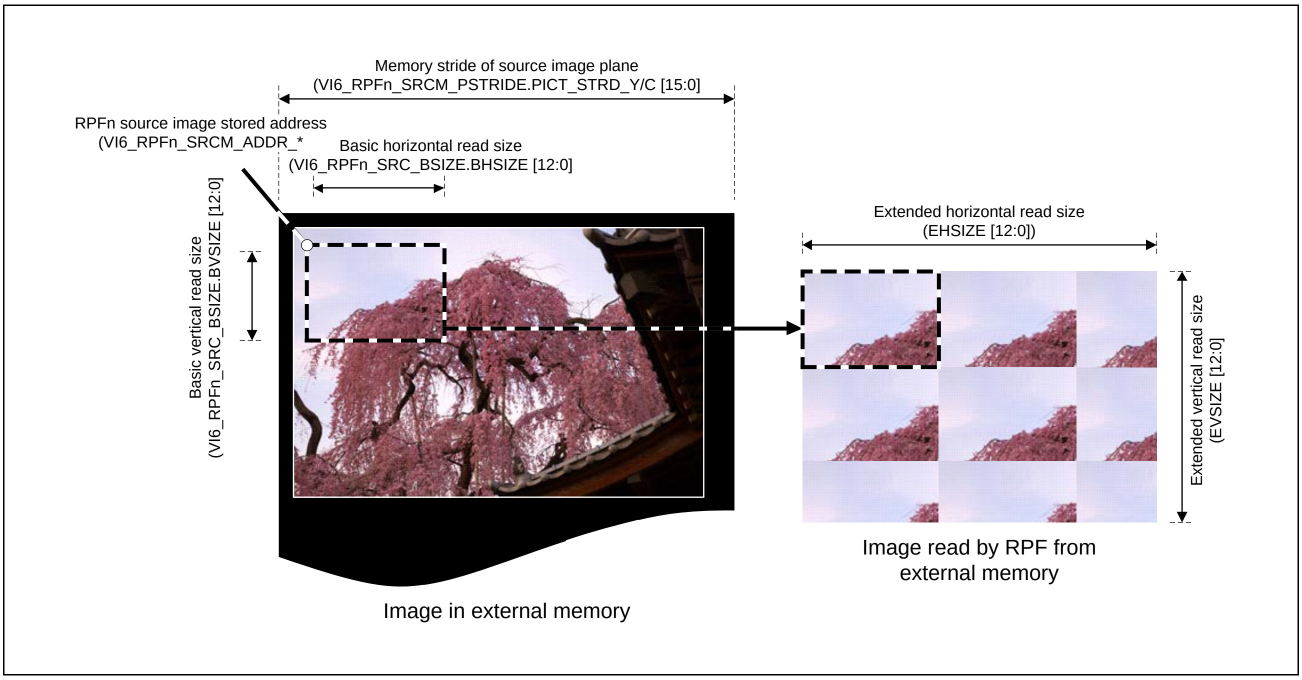


Figure 33.6 Relationship between Basic Read Size and Extended Read Size

Refer also to the following sections.

- **Section 33.3.2.6(2), RPFn Extended Read Size Registers (VI6_RPFn_SRC_ESIZE)**
- **Section 33.3.2.6(13), RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)**
- **Section 33.3.2.6(15), RPFn Source Y/RGB Address Registers (VI6_RPFn_SRCM_ADDR_Y)**
- **Section 33.3.2.6(16), RPFn Source Chroma Address Registers 0 (VI6_RPFn_SRCM_ADDR_C0)**
- **Section 33.3.2.6(17), RPFn Source Chroma Address Registers 1 (VI6_RPFn_SRCM_ADDR_C1)**

(2) RPFn Extended Read Size Registers (VI6_RPFn_SRC_ESIZE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	EHSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	EVSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	BHSIZE[12:0]	All 0	R/W	RPF Extended Horizontal Read Size These bits specify the horizontal size of the extended read area to which the RPFn reads data from the external RAM. As shown in **Figure 33.6**, the basic read area image is repeatedly placed in the extended read area; in the EHSIZE bits, specify a value not smaller than the horizontal size of the basic read area. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 1920 can be specified.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	EVSIZE[12:0]	All 0	R/W	RPF Extended Vertical Read Size These bits specify the vertical size of the extended read area to which the RPFn reads data from the external RAM. As shown in **Figure 33.6**, the basic read area image is repeatedly placed in the extended read area; in the EVSIZE bits, specify a value not smaller than the vertical size of the basic read area. When the input format is YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 1080 can be specified.

VI6_RPFn_SRC_ESIZE specifies the extended size for RPFn. The extended horizontal and vertical sizes should be equal to or greater than the basic sizes specified in VI6_RPFn_SRC_BSIZE. The RPF internal data processing described later and image processing described in **Section 33.3.2.8** and later sections are all applied to the image in the extended read size shown on the right side in **Figure 33.6**.

(3) RPFn Input Format Registers (VI6_RPFn_INFMT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VIR	—	—	—	—	—	—	—	—	—	—	—	CIPM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPYCS	SPUVS	CEXT[1:0]		RDTM[2:0]			CSC	—	RDFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	VIR	0	R/W	Virtual Input Enable Enables or disables the virtual input function of the RPFn. The image to be processed by the RPFn is usually read from the external memory by the MAU. Instead of this input, the virtual input function generates a single-color image within the RPFn and sends it to the modules in VSPD. When the virtual input function is enabled, the fixed value specified in VI6_RPFn_VRTCOL_SET is used as the input to the RPFn. While the virtual input function is enabled, data is not read from the external memory; that is, the α plane is not read and the IROP calculation thus cannot be executed. In this case, set VI6_RPFn_ALPH_SEL.ASEL to 4. Neither the color space conversion through CSC nor the color keying described in **Section 33.3.2.6(11)** can be used. 0: RPFn uses general input. 1: RPFn uses virtual input.
27 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CIPM	0	R/W	Horizontal Chrominance Interpolation Method Setting Image data is processed in the YCbCr444 format inside VSPD in case of YCbCr color space. When the chrominance format of the input image is YCbCr422 or YCbCr420, data is up-sampled as shown in **Figure 33.7** for internal processing. This bit specifies the method of up-sampling for this purpose. 0: The nearest-neighbor method is used for horizontal chrominance interpolation. 1: The bilinear method is used for horizontal chrominance interpolation.
15	SPYCS	0	R/W	RPF Input Mode Setting 1 When the input format is YUY2, set this bit to 1 and set the RDFMT bits to 71 (H’47). When the input format is YVYU, set this bit and the SPUVS bit to 1 and set the RDFMT bits to 71 (H’47). In other cases, set this bit to 0.
14	SPUVS	0	R/W	RPF Input Mode Setting 2 When the input format is NV61, set this bit to 1 and set the RDFMT bits to 65 (H’41). When the input format is NV21, set this bit to 1 and set the RDFMT bits to 66 (H’42). When the input format is YVYU, set this bit and the SPYCS bit to 1 and set the RDFMT bits to 71 (H’47). In other cases, set this bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
13, 12	CEXT[1:0]	All 0	R/W	Lower-Bit Color Data Extension Method Setting When an RGB input format where each color component is expressed in less than 8 bits are selected from **Table 33.12** through the RDFMT bits, VSPD internally extends each color component to 8 bits before using the data. These bits select this extension method. 00b: Lower-order bits of color data are extended with 0. 01b: Upper-order bits of color data are copied to the lower -order bits. 10b: Lower-order bits of color data are extended with 0. The maximum value is limited to H'FF. 11b: Setting prohibited
11 to 9	RDTM[2:0]	All 0	R/W	CSC Conversion Expression Setting These bits select the expression used for color space conversion. The conversion direction is RGB → YCbCr when RGB is selected through the RDFMT bits; the direction is YCbCr → RGB when YCbCr is selected. 0: BT.601 YCbCr [16,235/240] ↔ RGB [0,255] 1: BT.601 YCbCr [0,255] ↔ RGB [0,255] 2: BT.709 YCbCr [16,235/240] ↔ RGB [0,255] 3: BT.709 YCbCr [16,235/240] ↔ RGB [16,235] 4 to 7: Setting prohibited
8	CSC	0	R/W	Color Space Conversion Enable Enables or disables color space conversion between YCbCr and RGB to be executed in RPFn. The characteristics of color space conversion are determined by the RDTM bit setting.*1 When using the virtual input (VIR = 1), specify 0. 0: Color space conversion is disabled. 1: Color space conversion is enabled.
7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 0	RDFMT[6:0]	All 0	R/W	RPF Input Image Format Setting These bits select the format of the image input from the external RAM to the RPFn. Select a value corresponding to the desired format from those shown in **Table 33.12**, **Table 33.13**, and **Table 33.14**. When the virtual input function is used (VIR = 1), the color information for the virtual input should be specified in VI6_RPFn_VRTCOL_SET. If this color information is in the RGB format, set the RDFMT bits to 19. If the color information is in the YCbCr format, set these bits to 64. [Note 1] Number of input pixels When YCbCr4:2:2 is selected through the RDFMT bits, the horizontal size of the input image should be specified in 2-pixel units. When YCbCr4:2:0 is selected, the vertical and horizontal sizes should be specified in 2-pixel units. Observe these restrictions when specifying the image size in VI6_RPFn_SRC_BSIZE and VI6_RPFn_SRC_ESIZE.

Note 1. Note on color space settings

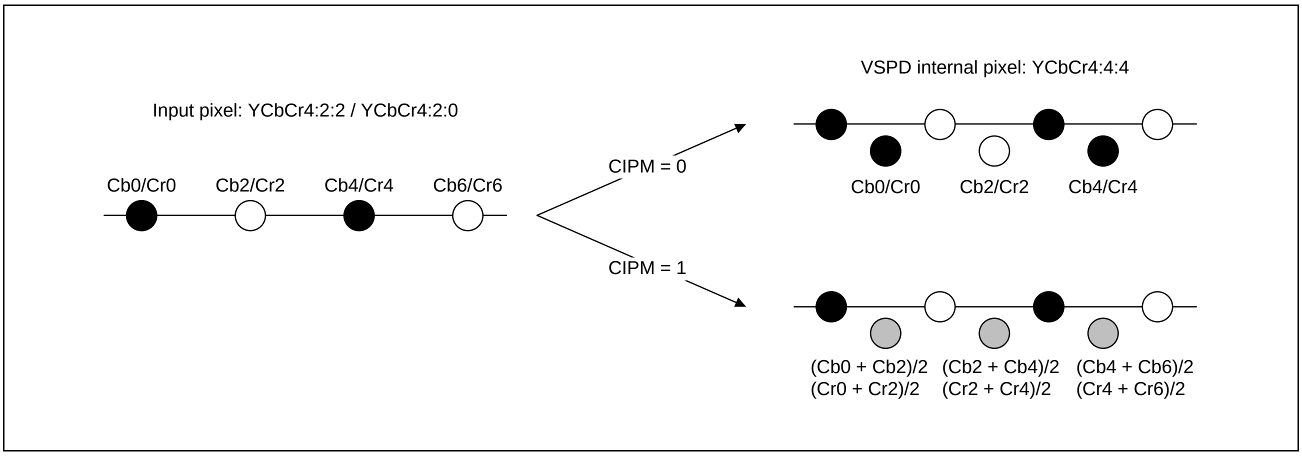


Figure 33.7 Chrominance Interpolation Methods Selectable through CIPM Setting

The color space for the image output from the RPF to VSPD internal modules is determined by the combination of the color space for the image input to the RPF, which is selected through the VI6_RPFn_INFMT.RDFMT setting, and the enabled or disabled state of the color space conversion function, which is selected through the VI6_RPFn_INFMT.CSC setting (**Table 33.11**). For example, when the image input to the RPF is in the YCbCr format, the RPF outputs data to VSPD internal modules in the YCbCr format if color space conversion is disabled through the CSC bit, and the RPF outputs data in the RGB format if color space conversion is enabled. When the image input to the RPF is in the RGB format, the relationship between the output format and the color space conversion setting is the opposite of the YCbCr case. For some VSPD internal modules, the YCbCr format is recommended for image processing because of the characteristics of the processing, or the same color space needs to be specified between multiple RPF outputs. In these cases, set VI6_RPFn_INFMT.RDFMT and VI6_RPFn_INFMT.CSC appropriately so that the RPFs can output the required color space according to the color space conditions described above.

Table 33.11 RPFn Input Color Space and Output Color Space

RPFn Input Color Space (VI6_RPFn_INFMT. RDFMT)		Color Space Conversion Setting (VI6_RPFn_INFMT.CSC)		RPFn Output Color Space
RGB	(H'00 to H'3F)*1	Disabled	(0) *1	RGB
		Enabled	(1) *1	YCbCr
YCbCr	(H'40 to H'7F)*1	Disabled	(0) *1	YCbCr
		Enabled	(1) *1	RGB

Note 1. Value specified in the register

A color space conversion function equivalent to that in the RPFn is also provided by the WPF. As shown in **Table 33.11**, the color space (YCbCr or RGB) output from the RPF becomes the input format for the WPF. Here, the color space of the output image obtained by the color space conversion function of the WPF must match the color space of the format specified through VI6_WPFn_OUTFMT.WRFMT.

Figure 33.8 shows the relationship between the input/output format and color space. The input color space for the RPF is determined when the input image format for the RPF is specified through the RDFMT bits. The color space for the image output from the RPF to subsequent VSPD internal modules depends on the combination of the RPF input format and CSC (color space conversion function) enabled or disabled state in the RPF as shown in **Table 33.11** and **Figure 33.8**. The user should first determine whether the image processing in the VSPD is done in YCbCr or RGB, and then specify the RPF input format and CSC enabled or disabled state to obtain the desired color space. The color space of the RPF output image is also that of the WPF input image; the color space of the data output from the WPF to the outside of

VSPD depends on the combination of the WPF input color space and the enabled or disabled state of the CSC implemented in the WPF as shown in **Figure 33.8.** The color space of the WPF output image must match that of the WPF output format (determined by VI6_WPFn_OUTFMT.WRFMT). For example, in the flow shown in **Figure 33.8.**, YCbCr should not be specified as the WPF output format regardless of the fact that the color space of the WPF output image is in RGB format.

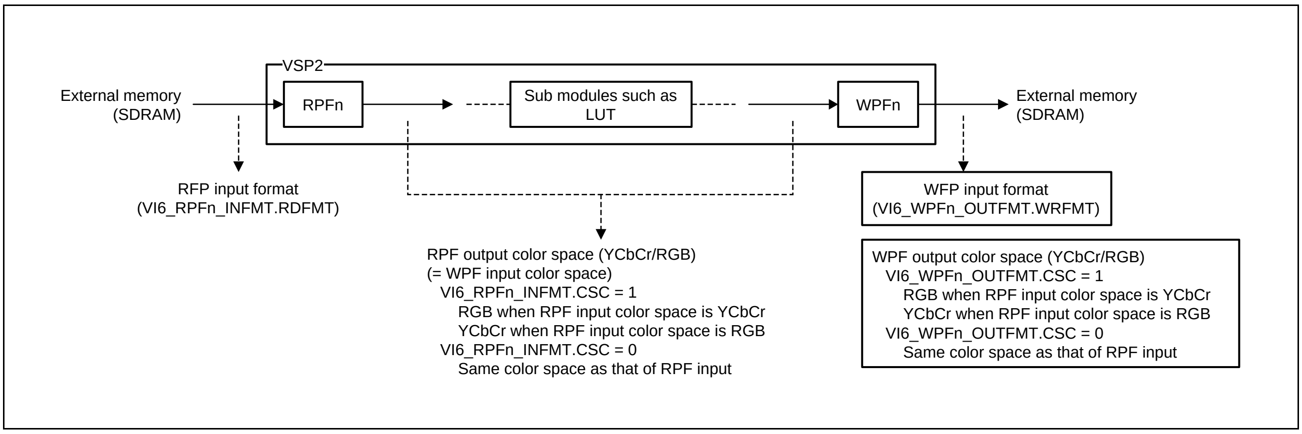


Figure 33.8 Relationship between Input/output Format and Color Space

Table 33.12 Packed Formats for RPF Input

			upper row - address / bottom row - bit field																															
			n								n+1								n+2								n+3							
RDFMT[6:0]	Bit per pixel	Phase	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
H'00	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
H'01	12	—					R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0					R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
H'02			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0					R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1				
H'03	—	—	Reserved								Reserved								Reserved								Reserved							
H'04	15	—		R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0		R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'05			R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0		R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	
H'06	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'07	18	—	A	A	A	A	A	A	A	A							R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
H'08			R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							A	A	A	A	A	A	A	A
H'09									R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
H'0A			A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0						
H'0B			A	A	A	A	A	A	A	A			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0
H'0C					R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
H'0D			A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0		
H'0E			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			A	A	A	A	A	A	A	A
H'0F		0							R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							R1	R1
		1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1							R2	R2	R2	R2	R2	R2	G2	G2	G2	G2
		2	G2	G2	B2	B2	B2	B2	B2	B2							R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3
H'10		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							R1	R1	R1	R1	R1	R1	G1	G1
		1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1							R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
		2	B2	B2							R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3						
H'11		0			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			R1	R1	R1	R1	R1	R1
		1			G1	G1	G1	G1	G1	G1			B1	B1	B1	B1	B1	B1			R2	R2	R2	R2	R2	R2			G2	G2	G2	G2	G2	G2
		2			B2	B2	B2	B2	B2	B2			R3	R3	R3	R3	R3	R3			G3	G3	G3	G3	G3	G3			B3	B3	B3	B3	B3	B3
H'12		0	R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			R1	R1	R1	R1	R1	R1		
		1	G1	G1	G1	G1	G1	G1			B1	B1	B1	B1	B1	B1			R2	R2	R2	R2	R2	R2			G2	G2	G2	G2	G2	G2		
		2	B2	B2	B2	B2	B2	B2			R3	R3	R3	R3	R3	R3			G3	G3	G3	G3	G3	G3			B3	B3	B3	B3	B3	B3		
H'13	24	—	A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
H'14			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
H'15		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
H'16	18	—								R0	R0	R0	R0	R0	R0	G0	G0	G0								G0	G0	G0	B0	B0	B0	B0	B0	B0
H'17																	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
H'18	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
H'19	12	—	A	A	A	A	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	A	A	A	A	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
H'1A		—	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	A	A	A	A	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	A	A	A	A
H'1B	15	—	A	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'1C		—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	A
H'1D	12	—	A	A	A	A	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	A	A	A	A	B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1
H'1E		—	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	A	A	A	A	B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1	A	A	A	A
H'1F	15	—	A	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	A	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1
H'20		—	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	A	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	A
H'21	18	0			B0	B0	B0	B0	B0	B0			G0	G0	G0	G0	G0	G0			R0	R0	R0	R0	R0	R0			B1	B1	B1	B1	B1	B1
		1			G1	G1	G1	G1	G1	G1			R1	R1	R1	R1	R1	R1			B2	B2	B2	B2	B2	B2			G2	G2	G2	G2	G2	G2
		2			R2	R2	R2	R2	R2	R2			B3	B3	B3	B3	B3	B3			G3	G3	G3	G3	G3	G3			R3	R3	R3	R3	R3	R3
H'22	24	—	A	A	A	A	A	A	A	A	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
H'23	16	—																	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0
H'24 to H'2F	—	—	Reserved								Reserved								Reserved								Reserved							

RDFMT[6:0]	Bit per pixel	Phase	upper row - address / bottom row - bit field																															
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
H'30	*1	*1	*1								*1								*1								*1							
H'31 to H'3F	—	—	Reserved								Reserved								Reserved								Reserved							

Note 1. When the RDFMT[6:0] bits are set to H'30, refer to **Table 33.13** for the additional settings and the packed formats

Table 33.13 Packed formats of RGB10, RGB10A2, A2RGB10

RDFMT[6:0]	Format name	BYPP_M1[2:0] CPOS[31:0] CLEN[31:0]	upper row - address / bottom row - bit field																															
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
H'30	RGB10	H'3 H'000A1400 H'0A0A0A00	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	B0	B0		
H'30	RGB10A2	H'3 H'000A141E H'0A0A0A02	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	B0	B0	A0	A0
H'30	A2RGB10	H'3 H'020C1600 H'0A0A0A02	A0	A0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	B0	B0

Table 33.14 Packed YCbCr Formats for RPF Input

RDFMT[6:0]	Packed YCbCr Input Format	Reference
H'40	YCbCr4:4:4 semi-planar	**Figure 33.9***4
H'41	YCbCr4:2:2 semi-planar (NV16, NV61*1)	
H'42	YCbCr4:2:0 semi-planar (NV12, NV21*1) *5	
H'43 to H'45	Reserved	—
H'46	YCbCr4:4:4 interleaved	**Figure 33.10***4
H'47	YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3)	
H'48	YCbCr4:2:2 interleaved type 1	
H'49	YCbCr4:2:0 interleaved *6	
H'4a	YCbCr4:4:4 planar	**Figure 33.11***4
H'4b	YCbCr4:2:2 planar (YV16)	
H'4c	YCbCr4:2:0 planar (YV12, I420) *5	
H'4d to H'7e	Reserved	—
H'7f	Reserved	—

Note 1. When the input format is NV61 or NV21, set the SPUVS bit to 1.

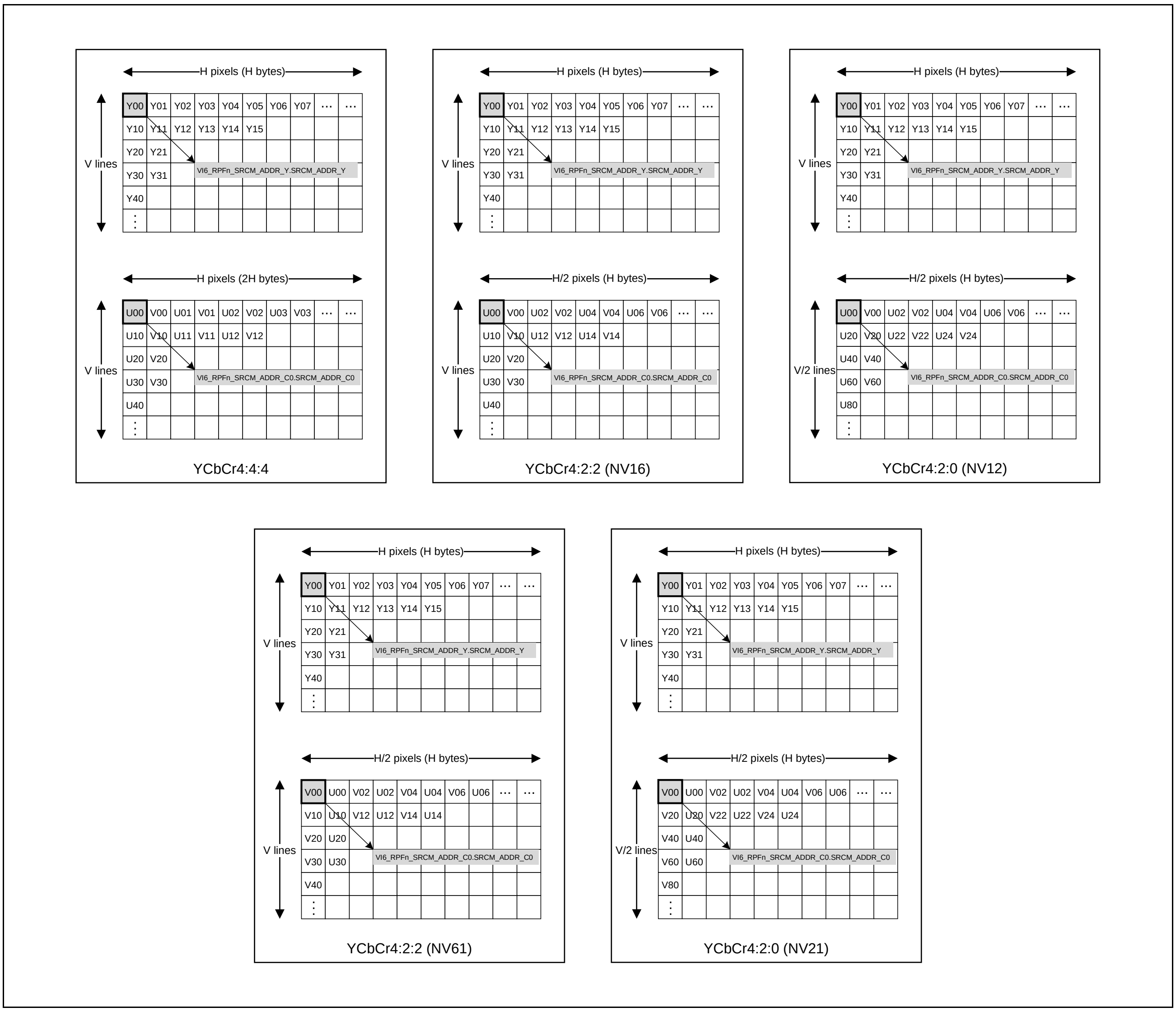
Note 2. When the input format is YUY2, set the SPYCS bit to 1.

Note 3. When the input format is YVYU, set the SPUVS bit to 1 and SPYCS bit to 1.

Note 4. **Figure 33.12** shows the definition of memory address for each pixel in **Figure 33.9** to **Figure 33.11**.

Note 5. Each line of C plane is read twice, so byte/pixel of YCbCr420 is 2 byte/pixel (same as YCbCr422).

Note 6. Each line of plane is read twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).



YCbCr4:4:4

YCbCr4:2:2 (NV16)

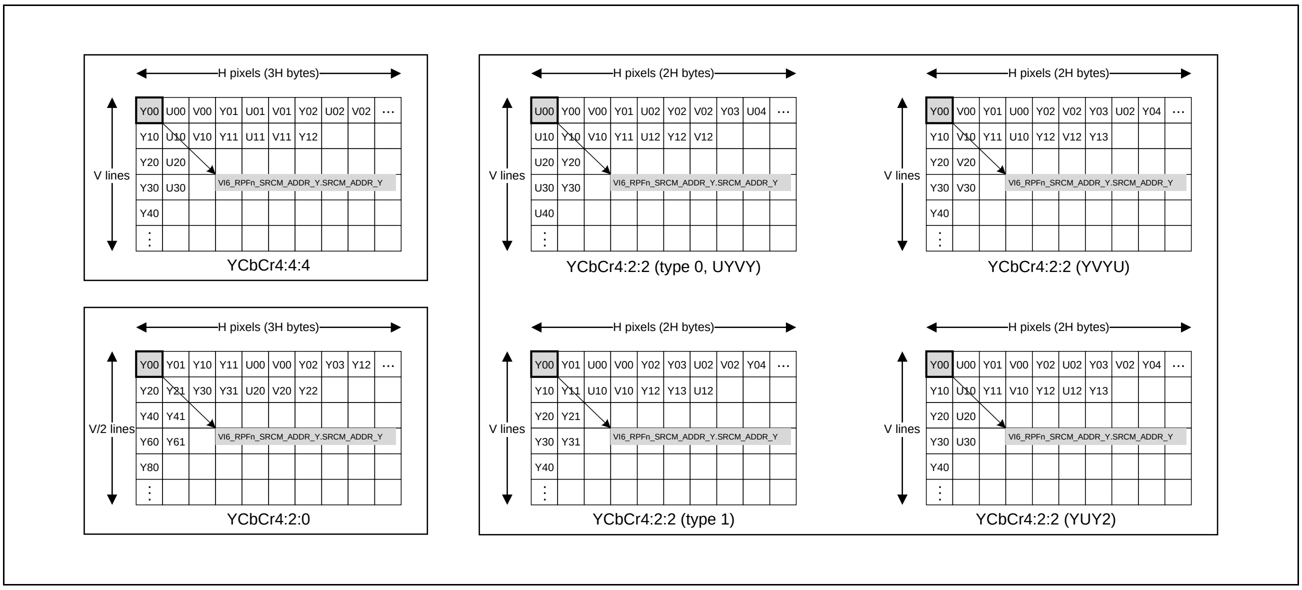
YCbCr4:2:0 (NV12)

YCbCr4:2:2 (NV61)

YCbCr4:2:0 (NV21)

Figure 33.9 YCbCr Semi-Planar Formats*1

Note 1. This figure is for 8bpc.



YCbCr4:4:4

YCbCr4:2:2 (type 0, UYVY)

YCbCr4:2:2 (YVYU)

YCbCr4:2:0

YCbCr4:2:2 (type 1)

YCbCr4:2:2 (YUY2)

Figure 33.10 YCbCr Interleaved Formats*[1]

Note 1. This figure is for 8bpc.

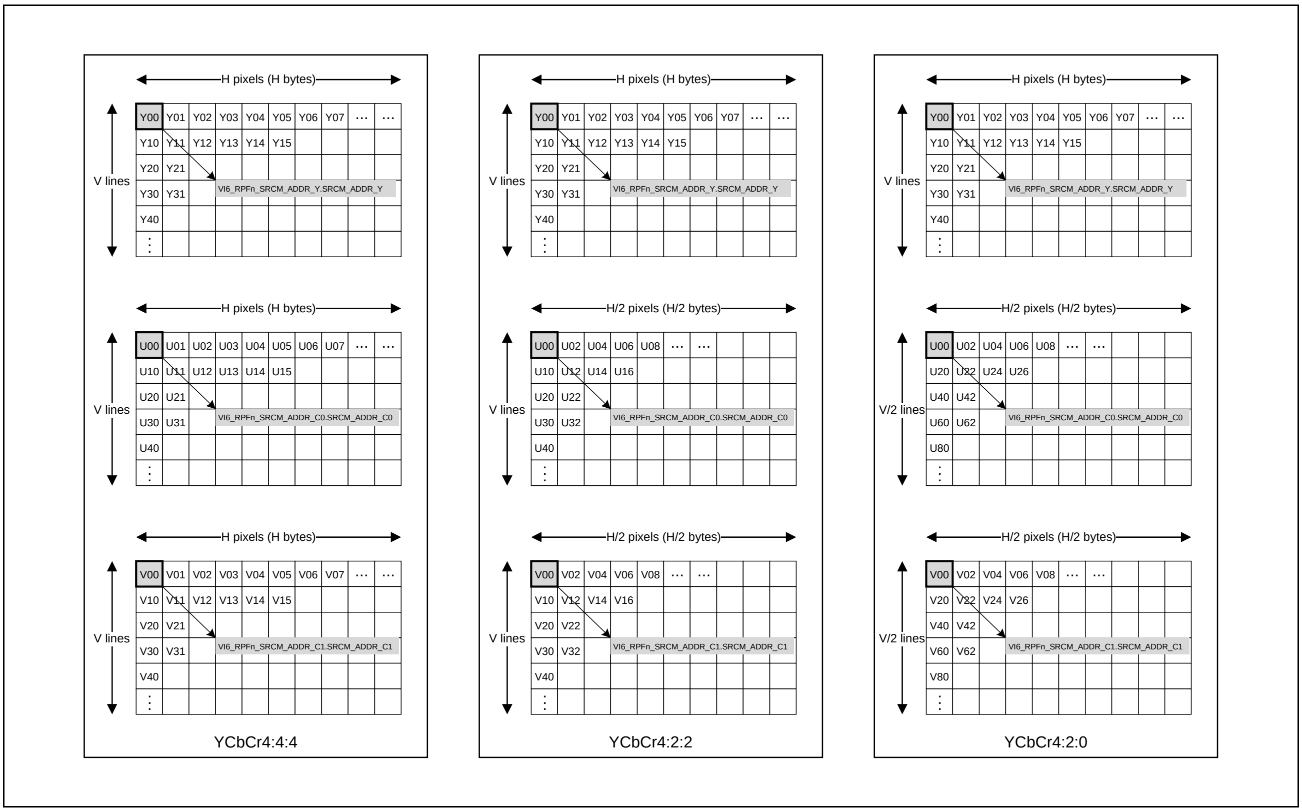
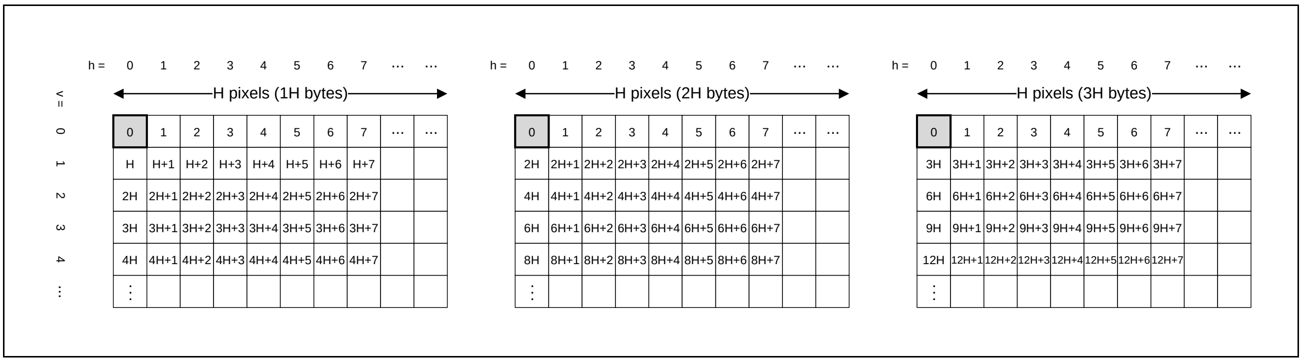


Figure 33.11 YCbCr Planar Formats*[1]

Note 1. This figure is for 8bpc.



v =	h = 0	1	2	3	4	5	6	7	…	…
0	0	1	2	3	4	5	6	7	…	…
1	H	H+1	H+2	H+3	H+4	H+5	H+6	H+7		
2	2H	2H+1	2H+2	2H+3	2H+4	2H+5	2H+6	2H+7		
3	3H	3H+1	3H+2	3H+3	3H+4	3H+5	3H+6	3H+7		
4	4H	4H+1	4H+2	4H+3	4H+4	4H+5	4H+6	4H+7		
⋮	⋮									

v =	h = 0	1	2	3	4	5	6	7	…	…
0	0	1	2	3	4	5	6	7	…	…
1	2H	2H+1	2H+2	2H+3	2H+4	2H+5	2H+6	2H+7		
2	4H	4H+1	4H+2	4H+3	4H+4	4H+5	4H+6	4H+7		
3	6H	6H+1	6H+2	6H+3	6H+4	6H+5	6H+6	6H+7		
4	8H	8H+1	8H+2	8H+3	8H+4	8H+5	8H+6	8H+7		
⋮	⋮									

v =	h = 0	1	2	3	4	5	6	7	…	…
0	0	1	2	3	4	5	6	7	…	…
1	3H	3H+1	3H+2	3H+3	3H+4	3H+5	3H+6	3H+7		
2	6H	6H+1	6H+2	6H+3	6H+4	6H+5	6H+6	6H+7		
3	9H	9H+1	9H+2	9H+3	9H+4	9H+5	9H+6	9H+7		
4	12H	12H+1	12H+2	12H+3	12H+4	12H+5	12H+6	12H+7		
⋮	⋮									

Figure 33.12 Memory address for the corresponding pixel's position

Legend of **Figure 33.12**

- H is horizontal image size in pixel unit.
- 1H, 2H and 3H is corresponding to the case 1 pixel has 1byte, 2bytes, 3bytes of data respectively.

(4) RPFn Data Swapping Registers (VI6_RPFn_DSWAP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	A_LLS	A_LWS	A_WDS	A_BTS	—	—	—	—	P_LLS	P_LWS	P_WDS	P_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	A_LLS	0	R/W	α Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled
10	A_LWS	0	R/W	α Plane Data Swapping in long word Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled
9	A_WDS	0	R/W	α Plane Data Swapping in Word Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in Word (16-bit) units is disabled 1: Data swapping in Word (16-bit) units is enabled
8	A_BTS	0	R/W	α Plane Data Swapping in Byte Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in Byte (8-bit) units is disabled 1: Data swapping in Byte (8-bit) units is enabled
7 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	P_LLS	0	R/W	Picture Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.
2	P_LWS	0	R/W	Picture Plane Data Swapping in long word Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.

Bit	Bit Name	Initial Value	R/W	Description
1	P_WDS	0	R/W	Picture Plane Data Swapping in Word Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in Word (16-bit) units is disabled 1: Data swapping in Word (16-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.
0	P_BTS	0	R/W	Picture Plane Data Swapping in Byte Units The effect of this bit setting is defined in **Table 33.15**. 0: Data swapping in Byte (8-bit) units is disabled 1: Data swapping in Byte (8-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.

When the virtual input function of the RPFn is used (VI6_RPFn_INFMT.VIR = 1), this register setting is ignored. Swapping of RPF input data can be specified separately for the α plane and picture plane.

Table 33.15 shows the data order before and after swapping according to the long long word, long word, word, and byte swapping settings.

When data order in memory for each format is the same as **Table 33.12** for RGB format and **Figure 33.9** to **Figure 33.12** for YCbCr format, set 1111b to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is different from the definition, change data order within 16byte unit by these bits as shown in **Table 33.15**.

Table 33.15 Changing data order according to swap register

Data order in memory																				
Byte address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15	*_LLS	*_LWS	*_WDS	*_BTS
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	1	1	1	1
	1	0	3	2	5	4	7	6	9	8	11	10	13	12	15	14	1	1	1	0
	2	3	0	1	6	7	4	5	10	11	8	9	14	15	12	13	1	1	0	1
	3	2	1	0	7	6	5	4	11	10	9	8	15	14	13	12	1	1	0	0
	4	5	6	7	0	1	2	3	12	13	14	15	8	9	10	11	1	0	1	1
	5	4	7	6	1	0	3	2	13	12	15	14	9	8	11	10	1	0	1	0
	6	7	4	5	2	3	0	1	14	15	12	13	10	11	8	9	1	0	0	1
	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	1	0	0	0
	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	0	1	1	1
	9	8	11	10	13	12	15	14	1	0	3	2	5	4	7	6	0	1	1	0
	10	11	8	9	14	15	12	13	2	3	0	1	6	7	4	5	0	1	0	1
	11	10	9	8	15	14	13	12	3	2	1	0	7	6	5	4	0	1	0	0
	12	13	14	15	8	9	10	11	4	5	6	7	0	1	2	3	0	0	1	1
	13	12	15	14	9	8	11	10	5	4	7	6	1	0	3	2	0	0	1	0
	14	15	12	13	10	11	8	9	6	7	4	5	2	3	0	1	0	0	0	1
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	0	0	0	0

Table 33.12 for RGB format and **Figure 33.9** to **Figure 33.12** for YCbCr format																
Byte address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(5) RPFn Display Location Registers (VI6_RPFn_LOC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	HCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	HCOORD [12:0]	All 0	R/W	Horizontal Coordinate of Sublayer Display Location on Master Layer These bits specify the left-end location of the sublayer displayed by the RPFn and the subsequent module connected through the DPR. Specify the horizontal coordinate of the location in pixel units with the left-end pixel of the master layer set at coordinate 0. When the RPFn is the master layer, set these bits to 0. If the sublayer extends beyond the master layer according to the HCOORD setting, the extended section is cut off at the right end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the right end of the master layer. A value from 0 to 1919 can be specified.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	VCOORD [12:0]	All 0	R/W	Vertical Coordinate of Sublayer Display Location on Master Layer These bits specify the top-end location of the sublayer displayed by the RPFn and the subsequent module connected through the DPR. Specify the vertical coordinate of the location in pixel units with the top-end pixel of the master layer set at coordinate 0. When the RPFn is the master layer, set these bits to 0. If the sublayer extends beyond the master layer according to the VCOORD setting, the extended section is cut off at the bottom end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the bottom end of the master layer. A value from 0 to 1079 can be specified.

Figure 33.13 shows an example of RPF1 offsets with respect to master layer RPF0. Although this figure only shows sublayers RPF1, specify offsets for all RPFs other than the master layer in the same way as shown in this example.

Whether an RPFn is the master layer or a sublayer is determined through the selection of the source RPF for WPFn (the VI6_WPFn_SRCRPF setting). For details, refer to **Section 33.3.2.7(1)**.

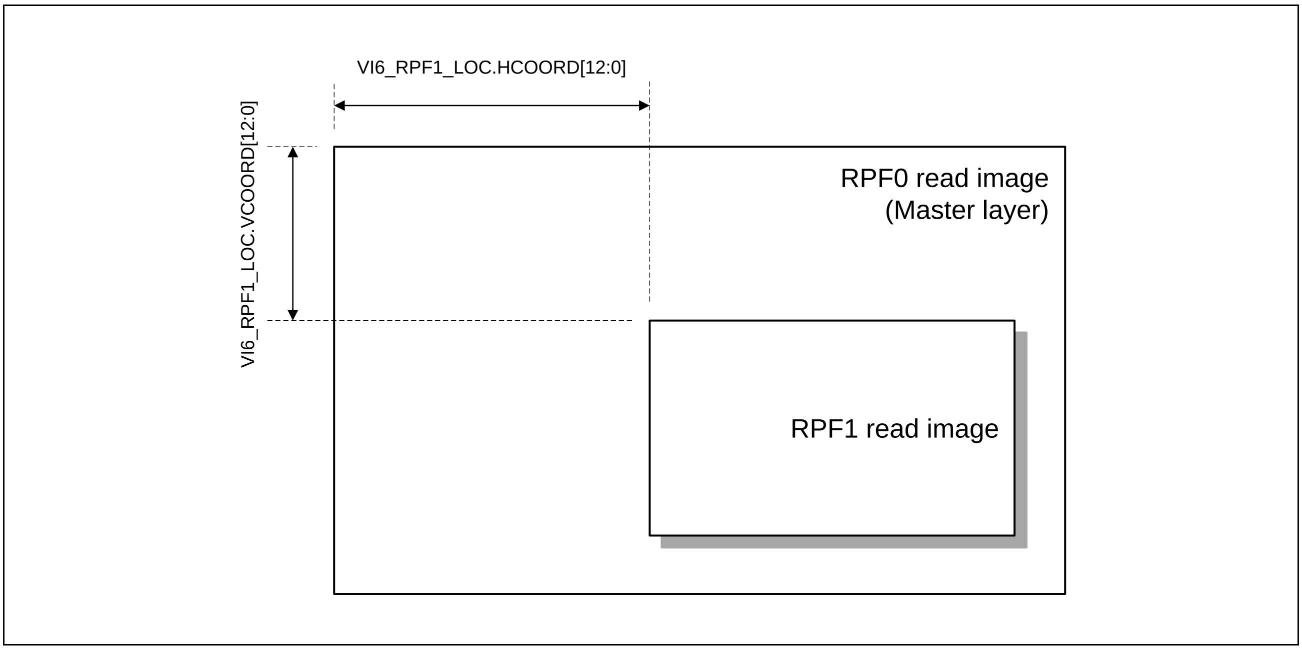


Figure 33.13 RPF1 Offsets from Master Layer

(6) RPFn α Plane Selection Control Registers (VI6_RPFn_ALPH_SEL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	ASEL[2:0]			IROP[3:0]				BSEL	—	—	—	AEXT[1:0]		—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPHA1[7:0]								ALPHA0 [7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
30 to 28	ASEL[2:0]	All 0	R/W	α Format and Processing Method Select These bits select how to handle the α value to be used. The RPF handles two types of α value; 8-bit and 1-bit values. When a 1-bit α value is used, VSPD assumes that the 1-bpp α value for each pixel is stored in the order from MSB to LSB in each byte (big endian). The α value is used as either transparency information or mask information. Transparency information is included in the α plane read from external memory when the ASEL bits are set to 1 or 3 and in the α value stored in the packed RGB bit field when these bits are set to 0 or 2. The α value as transparency information is sent as the destination value to the IROP as shown in **Figure 33.14** and then output to the subsequent modules. The output α value is used, for example, for blending in the BRS. The α value as mask information is used for IROP operation in the RPF. The mask information is included in the α plane read from external RAM when the ASEL bits are set to 0 or 2 and the source value is used in IROP operation (IROP setting other than 0, 5, 10, or 15). This α value is sent as the source value to the IROP as shown in **Figure 33.14**. Note that the α value selected through the ASEL bits has a lower priority than the VI6_RPFn_CKEY_SET*.AP* value replaced through the color keying function. When the color keying function is used, the α value may be replaced with the VI6_RPFn_CKEY_SET*.AP* value regardless of the ASEL bit setting. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), specify 4. 0: 1, 4, or 8-bit packed α + plane α (IROP ! = 0, 5, 10, 15) The α bit field in 1, 4, or 8-bit packed α is handled as transparency information. Be sure to specify the packed format that includes α through VI6_RPFn_INFMT.RDFMT. When VI6_RPFn_MSKCTRL.MSK_EN is 0 and the IROP bit value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPFn_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read. 1: 8-bit plane α The 8-bit α plane is read from external RAM as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPFn_SRCM_ADDR_AI) should be specified. The α value goes through the 8-bit transparent α generator shown in **Figure 33.14** without change. When VI6_RPFn_MSKCTRL.MSK_EN is 0, IROP operation cannot be executed; set the IROP bits to 0 in this case. When VI6_RPFn_MSKCTRL.MSK_EN is 1. IROP operation can be executed. 2: 1-bit packed α + plane α (IROP ! = 0, 5, 10, 15) The 1-bit packed α input is converted by the 8-bit transparent α generator shown in **Figure 33.14** according to the ALPHA0/1 setting into the 8-bit α value as transparency information. Select the packed input format that includes a 1-bit α field. When VI6_RPFn_MSKCTRL.MSK_EN is 0 and the IROP value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPFn_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read. 3: 1-bit plane α The 1-bit α plane is read from external RAM and converted by the 8-bit transparent α generator shown in **Figure 33.14** according to the ALPHA0/1 setting into the 8-bit α value as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPFn_SRCM_ADDR_AI) should be specified. When VI6_RPFn_MSKCTRL.MSK_EN is 0, IROP operation cannot be executed; set the IROP bits to 0 in this case. When VI6_RPFn_MSKCTRL.MSK_EN is 1. IROP operation can be executed. 4: Fixed α The fixed α value (VI6_RPFn_VRTCOL_SET.LAYA value) is output from the RPF. IROP operation cannot be executed; set the IROP bits to 0 in this case. 5 to 7: Setting prohibited.

Bit	Bit Name	Initial Value	R/W	Description
27 to 24	IROP [3:0]	All 0	R/W	IROP Operation Setting These bits specify the operator to be executed in the IROP operation unit shown in **Figure 33.14**. The source (S) for the IROP operation is the pixel data and α data specified in the VI6_RPFn_MSKSET0 or VI6_RPFn_MSKSET1 IROP input value register, which is selected according to the value (0 or 1) generated by the 1-bit mask generator. The destination (D) is the image data (RGB/YCbCr) and 8-bit α data output from the unpack/OSD processor. IROP operation is applied both for the image data and α data between the source and destination data. If these bits are set to the operation that involves the source (S) (IROP setting other than 0, 5, 10, or 15) while VI6_RPFn_MSKCTRL.MSK_EN is 0, the α plane is read from the external RAM to be used for the α value for IROP operation; specify the α plane read start address (VI6_RPFn_SRCM_ADDR_A). When the virtual input function is used (VI6_RPFn_INFMT.VIR = 1), IROP operation is not available; set these bits to 0000b. 0000b: NOP(D) 0001b: AND(S & D) 0010b: AND_REVERSE(S & ~D) 0011b: COPY(S) 0100b: AND_INVERTED(~S & D) 0101b: CLEAR(0) 0110b: XOR(S ^ D) 0111b: OR(S | D) 1000b: NOR(~(S | D)) 1001b: EQUIV(~(S ^ D)) 1010b: INVERT(~D) 1011b: OR_REVERSE(S | ~D) 1100b: COPY_INVERTED(~S) 1101b: OR_INVERTED(~S | D) 1110b: NAND(~(S & D)) 1111b: SET(all 1)
23	BSEL	0	R/W	α Bit Count Conversion Selection for 1-Bit Mask Generator Specifies the number of bits in the α plane to be read as mask information from the external RAM. The α value in mask information is used for the source (S) in IROP. When α plane data is 8 bits, it is converted to one bit through the 1-bit mask generator shown in **Figure 33.14**. Note that this bit setting is valid when the ASEL bits are set to 0 or 2 and VI6_RPFn_MSKCTRL.MSK_EN is set to 0. In other cases, this bit setting has no effect. 0: 8-bit α is converted to 1-bit α through the 1-bit mask generator. When the 8-bit α value input to the RPF is not 0, it is converted to 1b; when the value is 0, it is converted to 0b. 1: α value goes through the 1-bit mask generator. The 1-bit α value input to the RPF is output through the 1-bit mask generator without change.
22 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19 to 18	AEXT[1:0]	All 0	R/W	Lower-Bit α Value Extension Method Set These bits specify the method for extending the input α data to 8 bits through the unpack processing. 00b: The lower-order bits of α value are extended with 0. 01b: The upper-order bits of α value are copied to the lower-order bits. 10b: The lower-order bits of α value are extended with 0. The maximum value is limited to H'FF. 11b: Setting prohibited
17, 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	ALPHA1[7:0]	All 0	R/W	8-Bit α Value Output when 1-Bit α Value is 1 These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in **Figure 33.14** is 1b. This setting is valid when the ASEL bits are set to 010b or 011b. A value from 0 to 255 can be specified.
7 to 0	ALPHA0 [7:0]	All 0	R/W	8-Bit α Value Output when 1-Bit α Value is 0 These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in **Figure 33.14** is 0b. This setting is valid when the ASEL bits are set to 010b or 011b. A value from 0 to 255 can be specified.

Figure 33.14 shows the relationship between the α selector, IROP operation unit, color keying unit, and related registers. The IROP operation unit receives two inputs, source and destination. The image data input from the external memory is processed through the unpack processor and 8-bit transparent α generator and then input to the IROP operation unit as destination data. The α plane data input from the external memory is sent to the 8-bit transparent α generator when the ASEL bits are set to 1 or 3, or sent to the 1-bit mask α generator when the ASEL bits are set to 0 or 2. For the pixel data and 8-bit α value on the source side of the IROP operation unit, either the VI6_RPFn_MSKSET0 value or VI6_RPFn_MSKSET1 values will be selected according to the 1-bit α value output by the 1-bit mask α generator.

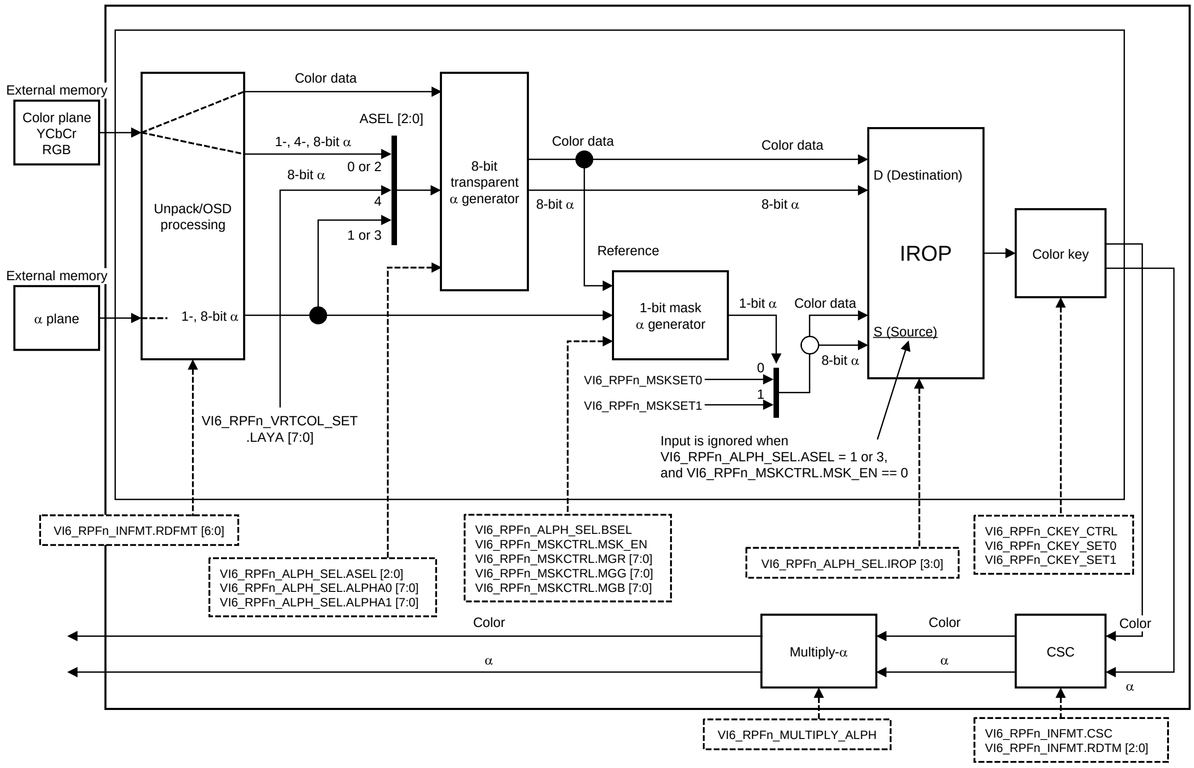


Figure 33.14 Configuration of alpha Selector and IROP Operation Unit in RPF

The following describes the function of each block shown in **Figure 33.14**. Read the following while referring to the figure as necessary.

Unpack/OSD processor:

Unpacks each component and α value of the image data according to the packed format specified in VI6_RPFn_INFMT.RDFMT.

8-bit transparent α generator:

Converts the input α value into 8-bit α when the input α is four bits or one bit.

When VI6_RPFn_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α), this generator outputs the input α value without change if the α bit field in the packed α data is 8 bits; if the α bit field is less than 8 bits, it is converted to an 8-bit α value by extending the LSB side according to the VI6_RPFn_ALPH_SEL.AEXT setting.

When VI6_RPFn_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 4 (fixed α), this generator outputs the input 8-bit plane α without change. If a packed α value is included in RGB data, it is discarded.

When VI6_RPFn_ALPH_SEL.ASEL are set to 2 (1-bit packed α) or 3 (8-bit α generated from 1-bit plane α), an 8-bit α value is generated by using the VI6_RPFn_ALPH_SEL.ALPHA0 [7:0] value when the input 1-bit α value is 0 or by using the VI6_RPFn_ALPH_SEL.ALPHA1 [7:0] value when the input 1-bit α value is 1. When VI6_RPFn_ALPH_SEL.ASEL is set to 3, a packed α value that is included in RGB data is discarded.

1-bit mask α generator:

Generates 1-bit α data from the input 8-bit α data or pixel data. When the input α data is one bit, this generator outputs it without change.

When VI6_RPFn_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α) or 2 (plane α) and VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the α plane read from the external memory to be used in IROP is converted to 1-bit α data when necessary. When the α plane data read from the external RAM is 8 bits (BSEL = 0), if the value is 0, a 1-bit α value of 0b is generated; if the value is not 0, a 1-bit α value of 1b is generated. When the α plane data is one bit (BSEL = 1), this generator outputs it without change.

When the value of the 1-bit α generated by the 1-bit mask α generator is 0b, the 8-bit α and pixel data specified in VI6_RPFn_MSKSET0 are output as the source. When the generated 1-bit α value is 1b, the 8-bit α and pixel data specified in VI6_RPFn_MSKSET1 are output as the source.

As shown in **Figure 33.14**, when VI6_RPFn_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 3 (1-bit plane α), the α plane read from the external RAM is sent to the 8-bit transparent α generator as transparency information. When VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the 1-bit α for masking is generated according to the input α plane (refer to **Section 33.3.2.6(8)**), but the 1-bit mask α generator does not refer to the input α plane because it is input to the 8-bit transparent α generator as transparency information. Accordingly, the 1-bit mask α generator does not generate a 1-bit α for masking and the data on the source side becomes invalid; that is, IROP operation cannot be executed. Set the IROP bits to 0 in this case. In contrast, when VI6_RPFn_MSKCTRL.MSK_EN is set to 1, the 1-bit mask α generator creates α data for masking according to the pixel data instead of the input α plane data, and IROP operation can be executed in this case.

IROP operation unit:

Executes ROP operation according to the opcode specified in VI6_RPFn_ALPH_SEL.IROP. For ROP operation (other than NOP), valid values should be input both for the source and destination. As described in the above (description of the 1-bit mask α generator), when VI6_RPFn_ALPH_SEL.ASEL are set to 1 or 3 and VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the source data for the IROP operation unit is treated as invalid; set VI6_RPFn_ALPH_SEL.IROP to 0 (NOP). When VI6_RPFn_ALPH_SEL.ASEL are set to 4, a fixed α value is output from the RPF and IROP operation is not available. In the same way as the above case, set VI6_RPFn_ALPH_SEL.IROP to 0 (NOP).

To specify a valid source value for the IROP operation unit and execute IROP operation (specify an opcode other than NOP in the IROP bits), specify register values as shown in **Table 33.16**. Where the source input state is indicated as "Valid" in the table, IROP operation can be executed. In the cases where IROP operation is not available, set the IROP bits to 0 (NOP).

Table 33.16 Source Input State in IROP Operation Unit

VI6_RPFn_ALPH_SEL.ASEL[2:0]		VI6_RPFn_MSKCTRL.MSK_EN		
		0 (Source data is generated according to input α plane)		1 (Source data is generated according to the destination-side pixel data)
000b	(1-, 4-, or 8-bit packed α + plane α)	Valid	(α plane input)	Valid
001b	(8-bit α plane)	Invalid	(IROP operation is not available; α plane is output to the subsequent modules behind RPF)	Valid
010b	(8-bit α generated from 1-bit packed α + plane α)	Valid	(α plane input)	Valid
011b	(8-bit α generated from 1-bit plane α)	Invalid	(IROP operation is not available; α plane is output to the subsequent modules behind RPF)	Valid
100b	(Fixed α)	Invalid (IROP operation is not available; fixed α is output to the subsequent modules behind RPF)		

For the handling of the α values shown in **Figure 33.14** and **Table 33.16**, the relationship between the RPF input format and RPF output α value is shown in **Table 33.17**. Where only bit names are shown in the table, the bits are in VI6_RPFn_ALPH_SEL described in this section.

Table 33.17 α Value Selected and Output according to ASEL Bits in Each Input Format

ASEL Setting		α Value Output for Each Input Format	
		RGB	YCbCr
000b	(8-, 4-, or 1-bit packed α is input)	1- ,4-, or 8-bit pixel α	H'FF*1
001b	(8-bit plane α is input)	8-bit α plane	8-bit α plane
010b	(8-bit α is generated from the 1-bit packed α input)	ALPHA0 or ALPHA1 setting	H'FF*1
011b	(8-bit α is generated from the 1-bit plane α input)	ALPHA0 or ALPHA1 setting	ALPHA0 or ALPHA1 setting
100b	(Fixed α is output)	VI6_RPFn_VRTCOL_SET.LAYA setting	

Note 1. Fixed value H'FF is output because packed α is not included in YCbCr.

(7) RPFn Virtual Plane Color Information Registers (VI6_RPFn_VRTCOL_SET)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LAYA[7:0]								LAYR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LAYG[7:0]								LAYB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	LAYA[7:0]	All 0	R/W	Virtual-Input Fixed α Value These bits specify the fixed α value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When the virtual input function is disabled (VI6_RPFn_INFMT.VIR = 0), these bits are used to specify the fixed α value to be output from the RPF while VI6_RPFn_ALPH_SEL.ASEL are set to 4. A value from 0 to 255 can be specified.
23 to 16	LAYR[7:0]	All 0	R/W	Virtual-Input Fixed R/Cr Component Value These bits specify the fixed R or Cr value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the R value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cr value. A value from 0 to 255 can be specified.
15 to 8	LAYG[7:0]	All 0	R/W	Virtual-Input Fixed G/Y Component Value These bits specify the fixed G or Y value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the G value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Y value. A value from 0 to 255 can be specified.
7 to 0	LAYB[7:0]	All 0	R/W	Virtual-Input Fixed B/Cb Component Value These bits specify the fixed B or Cb value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the B value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cb value. A value from 0 to 255 can be specified.

(8) RPFn Mask Control Registers (VI6_RPFn_MSKCTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	MSK_EN	MGR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MGG[7:0]								MGB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	MSK_EN	0	R/W	Mask Generation Specification Specifies the method of α value generation in the 1-bit mask α generator shown in **Figure 33.14**. 0: A 1-bit mask value is generated according to the input α plane value. When the input α is in the 1-bit format (VI6_RPFn_ALPH_SEL.BSEL = 1), the 1-bit mask value is output without change. When the input α is in the 8-bit format (VI6_RPFn_ALPH_SEL.BSEL = 0), the 1-bit mask value is 0 if the α value is H’00; otherwise, the 1-bit mask value is 1. 1: The R/Cr, G/Y, and B/Cb components of the image input to the destination side of the IROP operation unit are compared with the values specified in the MGR, MGG, and MGB bits, respectively. When all values match, 1 is output as the 1-bit mask value, and in other cases, 0 is output. When the generated 1-bit mask data is not used, set VI6_RPFn_ALPH_SEL.IROP to 0.
23 to 16	MGR[7:0]	All 0	R/W	R/Cr Comparison Value for 1-Bit α Generation These bits specify the R/Cr value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R value for comparison. When YCbCr is specified, specify a Cr value for comparison. This setting is ignored when the MSK_EN bit is set 0. A value from 0 to 255 can be specified.
15 to 8	MGG[7:0]	All 0	R/W	G/Y Comparison Value for 1-Bit α Generation These bits specify the G/Y value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G value for comparison. When YCbCr is specified, specify a Y value for comparison. This setting is ignored when MSK_EN is set to 0. A value from 0 to 255 can be specified.
7 to 0	MGB[7:0]	All 0	R/W	B/Cb Comparison Value for 1-Bit α Generation These bits specify the B/Cb value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B value for comparison. When YCbCr is specified, specify a Cb value for comparison. This setting is ignored when MSK_EN is set to 0. A value from 0 to 255 can be specified.

(9) RPFn IROP-SRC Input Value Registers 0 (VI6_RPFn_MSKSET0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSA0[7:0]								MSR0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSG0[7:0]								MSB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MSA0[7:0]	All 0	R/W	IROP-Source Input α Value when 1-Bit α is 0 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 33.14**). A value from 0 to 255 can be specified.
23 to 16	MSR0[7:0]	All 0	R/W	IROP-Source Input R/Cr Value when 1-Bit α is 0 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 33.14**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified.
15 to 8	MSG0[7:0]	All 0	R/W	IROP-Source Input G/Y Value when 1-Bit α is 0 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 33.14**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified.
7 to 0	MSB0[7:0]	All 0	R/W	IROP-Source Input B/Cb Value when 1-Bit α is 0 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 33.14**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified.

(10) RPFn IROP-SRC Input Value Registers 1 (VI6_RPFn_MSKSET1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSA1[7:0]								MSR1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSG1[7:0]								MSB1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MSA1[7:0]	All 0	R/W	IROP-Source Input α Value when 1-Bit α is 1 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1. A value from 0 to 255 can be specified (**Figure 33.14**).
23 to 16	MSR1[7:0]	All 0	R/W	IROP-Source Input R/Cr Value when 1-Bit α is 1 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 33.14**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified.
15 to 8	MSG1[7:0]	All 0	R/W	IROP-Source Input G/Y Value when 1-Bit α is 1 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 33.14**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified.
7 to 0	MSB1[7:0]	All 0	R/W	IROP-Source Input B/Cb Value when 1-Bit α is 1 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 33.14**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified.

(11) RPFn Color Keying Control Registers (VI6_RPFn_CKEY_CTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LTH	—	—	—	CV	—	—	SAPE1	SAPE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	LTH	0	R/W	Transparent color-luma Threshold Mode Enable/Disable This bit enables or disables transparent color-luma threshold mode for the color keying module. In transparent color-luma threshold mode, color information 0 (GY0) specified in VI6_RPFn_CKEY_SET0 register is compared with the input luma values. When the input data is in YCbCr format, and if input Y value is equal to or smaller than specified color information (VI6_RPFn_CKEY_SET0.GY0), the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET0.AP0. This bit is available only when the input data is in YCbCr format. When the input data is in an RGB format, set this bit to 0. This bit setting is valid only when the CV bit is set to 0; it is ignored when the CV bit is set to 1. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0. 0: Luma threshold mode is disabled (Matched color mode) 1: Luma threshold mode is enabled
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CV	0	R/W	Color Replacement Control This bit controls the color replacement function in the color keying module shown in **Figure 33.2**. When an RGB format is specified as the color space of the RPFn input data through VI6_RPFn_INFMT.RDFMT, and if all components of an input pixel match the color components specified in VI6_RPFn_CKEY_SET0, the color replacement function replaces the values of the input α and all RGB components with the α and color components specified in VI6_RPFn_CKEY_SET1. When a YCbCr format is specified as the color space of the RPFn input data through VI6_RPFn_INFMT.RDFMT, only the Y data is compared; if the luminance component of an input pixel matches the value specified in VI6_RPFn_CKEY_SET0.GY0, the color replacement function replaces the values of the input α and all YCbCr components with the α and color components specified in VI6_RPFn_CKEY_SET1. When the CV bit is set to 1, the color replacement function is enabled. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0. 0: Color replacement function is disabled (transparent color mode). 1: Color replacement function is enabled.
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1	SAPE1	0	R/W	Comparison Color Data Setting 1 Enable/Disable This bit enables or disables comparison color data setting 1 in the transparent color mode for the color keying module. This bit setting is valid only when the CV bit is set to 0; it is ignored when the CV bit is set to 1. In transparent color mode, color information 1 (VI6_RPFn_CKEY_SET1.R1/GY1/B1) specified in VI6_RPFn_CKEY_SET1 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET1.AP. When the input data is in YCbCr format, only the Y data is compared. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0. 0: Comparison color data setting 1 is disabled. 1: Comparison color data setting 1 is enabled.
0	SAPE0	0	R/W	Comparison Color Data Setting 0 Enable/Disable This bit enables or disables comparison color data setting 0 in the transparent color mode for the color keying module. This bit setting is valid only when the CV bit is set to 0; it is ignored when the CV bit is set to 1. In transparent color mode, color information 0 (VI6_RPFn_CKEY_SET0.R0/GY0/B0) specified in VI6_RPFn_CKEY_SET0 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET0.AP. When the input data is in YCbCr format, only the Y data is compared. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), clear this bit to 0. 0: Comparison color data setting 0 is disabled. 1: Comparison color data setting 0 is enabled.

(12) RPFn Color Keying Color Setting Registers m (VI6_RPFn_CKEY_SETm : m = 0, 1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	APm[7:0]								Rm[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GYm[7:0]								Bm[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	APm[7:0]	All 0	R/W	α Data in Color Keying Color Information-m ● In transparent color-matched color mode for color keying When the input data matches color setting-m (Rm, GYm, and Bm) for transparent color comparison in the color keying module, the input α value is replaced with the value specified in these bits. Specify the α value to replace the input value. ● In transparent color-luma threshold mode for color keying When the input data is in YCbCr format, and if input Y value is equal to or smaller than GY0, the input α value is replaced with the value specified in this bit field (AP0). Specify the α value in this bit field (AP0) to replace the input value. AP1 is not used in this mode. Set AP1 to 0 value. ● In color replacement mode for color keying These bits are not used in this mode. Clear them to 0. α value replacement through these bits in transparent color mode for color keying takes priority over the α value selected through the VI6_RPFn_ALPH_SEL.ASEL setting. A value from 0 to 255 can be specified.
23 to 16	Rm[7:0]	All 0	R/W	R*[1]/Cr Component Data in Color Keying Color Information-m ● In transparent color-matched color mode for color keying Specify the R component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPEm setting. When the RPFn input is in YCbCr format, the color keying module does not compare the Cr component, and the setting of these bits is ignored. ● In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. ● In color replacement mode for color keying Specify the R component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.
15 to 8	GYm[7:0]	All 0	R/W	G*[1]/Y Component Data in Color Keying Color Information-m ● In transparent color-matched color mode for color keying Specify the G/Y component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPEm setting. ● In transparent color-luma threshold mode for color keying Specify the Y component value in the GY0 to be compared. ● In color replacement mode for color keying Specify the G/Y component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	Bm[7:0]	All 0	R/W	B*[1]/Cb Component Data in Color Keying Color Information-m ● In transparent color-matched color mode for color keying Specify the B component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPEm setting. When the RPFn input is in YCbCr format, the color keying module does not compare the Cb component, and the setting of these bits is ignored. ● In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. ● In color replacement mode for color keying Specify the B component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Note 1. When comparison data is specified in an RGB format, if a packed format is selected for RGB input and each of the RGB components is not 8 bits, the lower-order bits of input data are extended as specified through VI6_RPFn_INFMT.CEXT before comparison. The RGB components to be compared with the input should also be extended in the same way and the extended values should be specified in this register.

(13) RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PICT_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PICT_STRD_Y[15:0]	All 0	R/W	Memory Stride of Source Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the source picture Y/RGB plane read by the RPFn. A value from 0 to 65535 can be specified. Refer to **Figure 33.15** for settings.
15 to 0	PICT_STRD_C[15:0]	All 0	R/W	Memory Stride of Source Picture C Plane These bits specify in 1-byte units the memory stride of the source picture C plane read by the RPFn. When an RGB-format picture is read, these bits do not need to be set. A value from 0 to 65535 can be specified. Refer to **Figure 33.15** for settings. In the YCbCr planar format, this setting is used as the memory stride of the Cb and Cr planes.

This register specifies the memory stride of the picture planes in the source area as shown in **Figure 33.15**. The memory stride of the α plane should be specified through VI6_RPFn_SRCM_ASTRIDE.ALPH_STRD. When the RPF input is in an RGB format, only the RGB plane is read; when the input is in YCbCr format, the Y and C planes are read as shown in **Figure 33.15**. When the α plane is used, the α plane is also read. According to the image format and the necessity of the α plane, specify the necessary addresses where the source image is stored (VI6_RPFn_SRCM_ADDR_Y, VI6_RPFn_SRCM_ADDR_C, and VI6_RPFn_SRCM_ADDR_AI).

Whether the α plane needs to be read is determined according to the α plane selection method and IROP operation type. For details, refer to **Section 33.3.2.6(6)**.

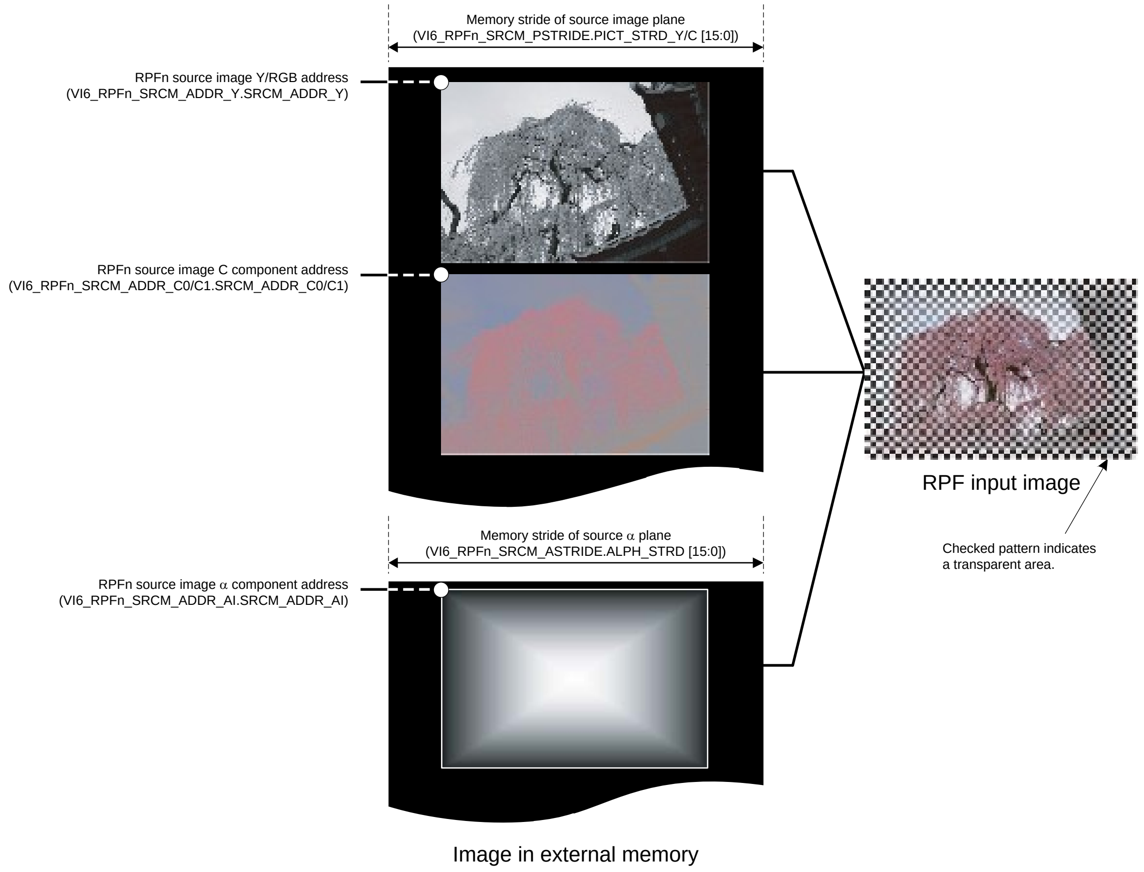


Figure 33.15 Reading an Image from RPFn Source Area

(14) RPFn Source α Memory Stride Setting Registers (VI6_RPFn_SRCM_ASTRIDE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPH_STRD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	ALPH_STRD [15:0]	All 0	R/W	Memory Stride of Source α Plane These bits specify in 1-byte units the memory stride of the source α plane read by the RPFn. A value from 0 to 65535 can be specified. Refer to **Figure 33.15** for settings.

(15) RPFn Source Y/RGB Address Registers (VI6_RPFn_SRCM_ADDR_Y)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_Y[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_Y[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR _Y[31:0]	All 0	R/W	Source Image Y/RGB Plane Storing Address These bits specify in 1-byte units the start address of the source image Y plane and packed RGB plane read by the RPFn. A value from H'0000 0000 to H'FFFF FFFF can be specified. Refer to **Figure 33.15** in **Section 33.3.2.6(13)** for settings.

(16) RPFn Source Chroma Address Registers 0 (VI6_RPFn_SRCM_ADDR_C0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_C0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_C0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_C0[31:0]	All 0	R/W	Source Image C Plane Storing Address 0 These bits specify in 1-byte units the start address of the source image C plane read by the RPFn. Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in **Figure 33.15** or the Cb plane when a planar format is selected. When an interleaved format is selected or the RPF input is in an RGB format, this setting is not used. A value from H’0000 0000 to H’FFFF FFFF can be specified. Refer to **Figure 33.15** in **Section 33.3.2.6(13)** for settings.

(17) RPFn Source Chroma Address Registers 1 (VI6_RPFn_SRCM_ADDR_C1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_C1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_C1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_C1[31:0]	All 0	R/W	Source Image C Plane Storing Address 1 These bits specify in 1-byte units the start address of the Cr plane when a planar YCbCr format shown in **Figure 33.15** is read by the RPFn. This setting is not used when the RPF input is in YCbCr format that is not a planar format or in an RGB format. A value from H’0000 0000 to H’FFFF FFFF can be specified. Refer to **Figure 33.15** in **Section 33.3.2.6(13)** for settings.

(18) RPFn Source α Address Registers (VI6_RPFn_SRCM_ADDR_AI)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_AI[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_AI[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_AI[31:0]	All 0	R/W	Source Image α Plane Storing Address These bits specify in 1-byte units the start address of the α plane of the source image read by the RPFn. Specify in the same way as the start address of the picture plane. When the α plane is not read from the source area, these bits do not need to be set. A value from H'0000 0000 to H'FFFF FFFF can be specified. Refer to **Figure 33.15** in **Section 33.3.2.6(13)** for settings.

(19) RPFn Bus Access Control Registers (VI6_RPFn_BAC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	B512
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	B512	0	R/W	Burst access in 512 pixels' enable 0: burst access in 256 pixels 1: burst access in 512 pixels Always set 0 to this bit
15 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

(20) RPFn Multiple Alpha Control Registers (VI6_RPFn_MULT_ALPH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	A_MMD	—	—	P_MMD[1:0]		ALPHA_RATIO[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12	A_MMD	0	R/W	0: Alpha data go through multiply-alpha unit. 1: Multiply-alpha unit multiplies alpha data by specified alpha (ALPHA_RATIO[7:0]). When output format from CSC unit is YCbCr, set 0 to this bit.
11, 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	P_MMD[1:0]	All 0	R/W	0: Image data go through multiply-alpha unit. 1: Multiply-alpha unit multiplies image data by specified alpha (ALPHA_RATIO[7:0]). 2: Multiply-alpha unit multiplies image data by alpha data. 3: Multiply-alpha unit multiplies image data by alpha data and specified alpha (ALPHA_RATIO[7:0]). When output format from CSC unit is YCbCr, set 0 to this bit.
7 to 0	ALPHA_RATIO[7:0]	All 0	R/W	Multiply alpha value

ALPin: Input Alpha data to Multiply-alpha unit

PIXin(R): Input R data to Multiply-alpha unit

PIXin(G): Input G data to Multiply-alpha unit

PIXin(B): Input B data to Multiply-alpha unit

ALPout: Output Alpha data from Multiply-alpha unit

PIXout(R): Output R data from Multiply-alpha

PIXout(G): Output G data from Multiply-alpha

PIXout(B): Output B data from Multiply-alpha

Table 33.18 Expression of output alpha data from Multiply-alpha unit

A_MMD	ALPHA_RATIO[7:0]	Expression
0	Don't care	ALPout = ALPin
1	Not 255	ALPout = ALPin * ALPHA_RATIO / 256
	255	ALPout = ALPin

Table 33.19 Expression of output pixel data from Multiply-alpha unit

P_MMD[1:0]	ALPHA_RATIO[7:0]	ALPin	Expression
0	Don't care	Don't care	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
1	Not 255	Don't care	PIXout(R) = PIXin(R) * ALPHA_RATIO / 256 PIXout(G) = PIXin(G) * ALPHA_RATIO / 256 PIXout(B) = PIXin(B) * ALPHA_RATIO / 256
	255	Don't care	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
2	Don't care	Not 255	PIXout(R) = PIXin(R) * ALPin / 256 PIXout(G) = PIXin(G) * ALPin / 256 PIXout(B) = PIXin(B) * ALPin / 256
	Don't care	255	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
3	Not 255	Not 255	PIXout(R) = PIXin(R) * ALPin * ALPHA_RATIO / 256 / 256 PIXout(G) = PIXin(G) * ALPin * ALPHA_RATIO / 256 / 256 PIXout(B) = PIXin(B) * ALPin * ALPHA_RATIO / 256 / 256
	255	Not 255	PIXout(R) = PIXin(R) * ALPin / 256 PIXout(G) = PIXin(G) * ALPin / 256 PIXout(B) = PIXin(B) * ALPin / 256
	Not 255	255	PIXout(R) = PIXin(R) * ALPHA_RATIO / 256 PIXout(G) = PIXin(G) * ALPHA_RATIO / 256 PIXout(B) = PIXin(B) * ALPHA_RATIO / 256
	255	255	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)

33.3.2.7 WPF Control Registers

(1) WPFn-Source-RPF Registers (VI6_WPFn_SRCRPF)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	VIR_ACT2[1:0]		—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RPF1_ACT[1:0]		RPF0_ACT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25, 24	VIR_ACT2 [1:0]	All 0	R/W	Virtual RPF Start Enable in BRS These bits enable start of the virtual RPF in the BRS as the source RPF for the WPF0 when the WPF0 is started. For details of the virtual RPF, refer to the following. **Section 33.3.2.10(1) BRS Input Control Register (VI6_BRS_INCTRL)** **Section 33.3.2.10(2) Size Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_SIZE)** **Section 33.3.2.10(3) Display Location Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_LOC)** **Section 33.3.2.10(4) Color Information Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_COL)** Note that the virtual RPF is in the BRS as shown in **Figure 33.28** and there are no register bits for DPR setting related to the virtual RPF. 0: The virtual RPF in the BRS is not started. 1: The virtual RPF in the BRS is started as a sublayer source RPF for the WPFn. 2: The virtual RPF in the BRS is started as the master-layer source RPF for the WPFn. 3: Setting prohibited
23 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3, 2	RPF1_ACT [1:0]	All 0	R/W	RPFn Start Enable (RPFn_ACT, n = 0 to 1) These bits enable start of RPFn as the source RPF for the WPFn when the WPFn is started. When RPFn is not started by any of the WPFn (n = 0), set the VI6_DPR_RPFn_ROUTE.RT_RPFn bits to D'63. 0: RPFn is not started. 1: RPFn is started as a sublayer source RPF for the WPFn. 2: RPFn is started as the master-layer source RPF for the WPFn. 3: Setting prohibited
1, 0	RPF0_ACT [1:0]	All 0	R/W	

When the WPFn is started through the VSPD start register n (VI6_CMDn: n = 0), the RPF and virtual RPF in the BRS specified as the source RPF in this register are also started to supply data to the VSPD internal modules.

Note the following when specifying the source RPF.

(1) When blending or ROP operation is applied to multiple images through the BRS, multiple source RPFs are necessary for one WPF. When multiple source RPFs are used, images should be classified into a master layer and sublayers; assign one of the source RPFs as the master-layer source RPF and other RPFs as sublayer source RPFs.

Do not assign all RPFs as sublayer source RPFs (VI6_WPF1_SRCRPF = H’00000015) or two or more RPFs as the master-layer source RPF (VI6_WPF0_SRCRPF = H’0000025A) (such settings are prohibited).

(2) When the BRS is not used, there should be only one source RPF for one WPF. In this case, the source RPF should be assigned as the master-layer source RPF.

(2) WPFn Horizontal Input Size Clipping Registers (VI6_WPFn_HSZCLIP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	HCEN	—	—	—	—	HCL_OFST[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HCL_SIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	HCEN	0	R/W	Horizontal Size Clipping Enable/Disable Enables or disables clipping of the horizontal size of the WPFn input image. 0: Horizontal size clipping is disabled 1: Horizontal size clipping is enabled
27 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	HCL_OFST [7:0]	All 0	R/W	Horizontal Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the left end of the image in horizontal size clipping when the HCEN bit is 1 (**Figure 33.16**). The left side of the image input to the WPF is cut off for the size specified in these bits. When the HCEN bit is 0, this setting is ignored. A value from 0 to 255 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in **Figure 33.16** is made, VSPD does not operate correctly.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	HCL_SIZE [12:0]	All 0	R/W	Horizontal Clipping Size Setting When the HCEN bit is 1, these bits specify the clipping size for horizontal clipping processing. Through this processing, the area of the horizontal size specified through the HCL_SIZE bits starting from the offset position specified through the HCL_OFST bits is determined as the valid image area. Accordingly, the right-side pixels beyond the (HCL_OFST + HCL_SIZE) size in the WPFn input image are discarded. When the HCEN bit is 0, this setting is ignored. A value from 1 to 1920 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in **Figure 33.16** is made, VSPD does not operate correctly. (Note) When the WPFn output format is YCbCr4:2:2 or YCbCr4:2:0, specify an even value in these bits.

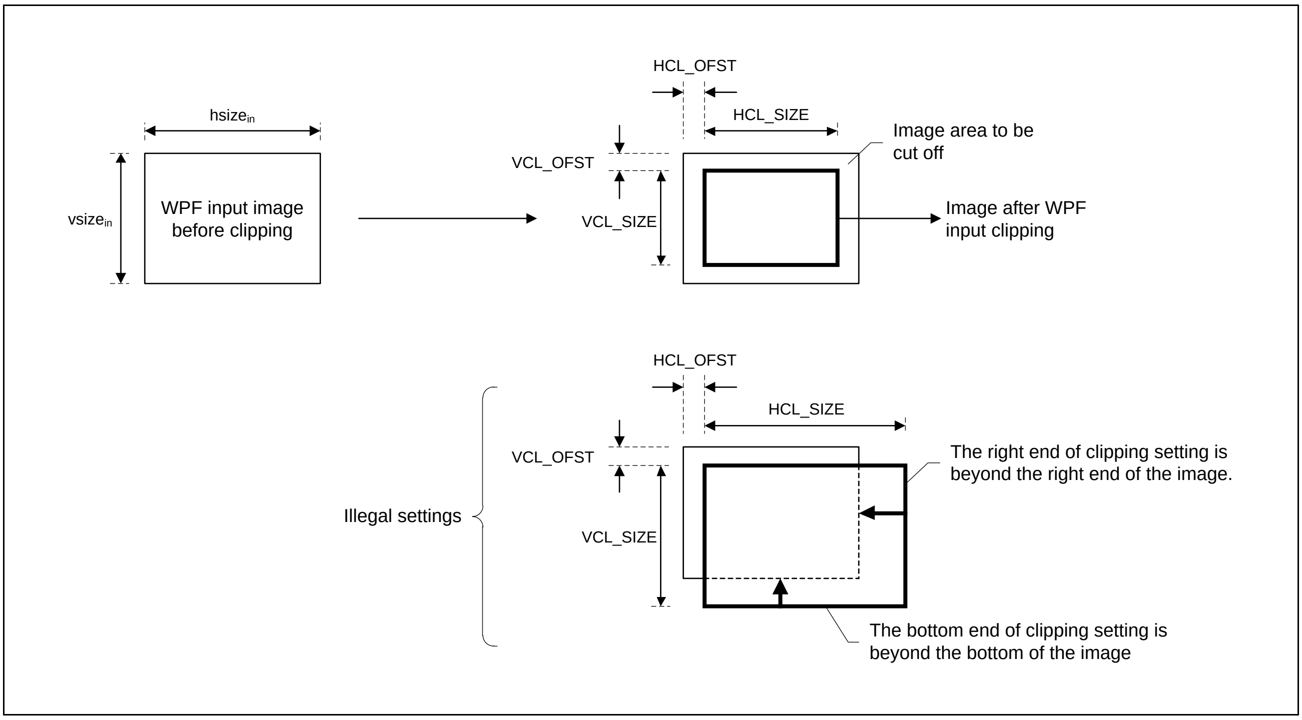


Figure 33.16 Image Clipping in WPF Input Section

(3) WPFn Vertical Input Size Clipping Registers (VI6_WPFn_VSZCLIP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VCEN	—	—	—	—	VCL_OFST[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VCL_SIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	VCEN	0	R/W	Vertical Size Clipping Enable/Disable Enables or disables clipping of the vertical size of the WPFn input image. 0: Vertical size clipping is disabled 1: Vertical size clipping is enabled
27 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	VCL_OFST [7:0]	All 0	R/W	Vertical Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the top end of the image in vertical size clipping when the VCEN bit is 1 (**Figure 33.16**). The top of the image input to the WPF is cut off for the size specified in these bits. When the VCEN bit is 0, this setting is ignored. A value from 0 to 255 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in **Figure 33.16** is made, VSPD does not operate correctly.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	VCL_SIZE [12:0]	All 0	R/W	Vertical Clipping Size Setting When the VCEN bit is 1, these bits specify the clipping size for vertical clipping processing. Through this processing, the area of the vertical size specified through the VCL_SIZE bits starting from the offset position specified through the VCL_OFST bits is determined as the valid image area. Accordingly, the bottom pixels beyond the (VCL_OFST + VCL_SIZE) size in the WPFn input image are discarded. When the VCEN bit is 0, this setting is ignored. A value from 1 to 1080 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in **Figure 33.16** is made, VSPD does not operate correctly. (Note) When the WPFn output format is YCbCr4:2:0, specify an even value in these bits.

(4) WPFn Output Format Registers (VI6_WPFn_OUTFMT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDV [7:0]								PXA	ODE	—	—	—	ROT[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPYCS	SPUVS	DITH[1:0]		WRTM[2:0]			CSC	—	WRFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PDV [7:0]	All 0	R/W	PAD Value in Output Packed Data These bits specify the value to be stored in the bit field indicated as PAD or P in the output formats shown in **Table 33.20**. To store this value in PAD, specify 0 in the PXA bit. A value from 0 to 255 can be specified.
23	PXA	0	R/W	PAD Data Select Selects the value to be stored in the bit field indicated as PAD or P in the packed RGB output formats shown in **Table 33.21**. Both the value specified in the PDV bits and the α data input from the DPR to WPF are 8 bits, but some of the PAD and P bit fields shown in **Table 33.20** are four bits or one bit. When the target bit field is not 8 bits, the number of bits in the PDV value and the α data input from the DPR to WPF is reduced according to the VI6_WPFn_RNDCTRL.ABRM setting. For bit count reduction, refer to **Figure 33.17** and the description of VI6_WPFn_RNDCTRL.ABRM. 0: The value specified in the PDV bits is stored in the PAD shown in **Table 33.20**. 1: The α value output from DPR in pixel units is stored in the PAD shown in **Table 33.20**.
22	ODE	0	R/W	Ordered Dither (mode A) Enable/Disable 0: Ordered dither (mode A) is disabled. 1: Ordered dither (mode A) is enabled. When the output format specified through the WRFMT bits is YCbCr, specify 0 in this bit. And when VI6_WPFn_OUTFMT.CSC is set to 1, specify 0 in this bit even in the case that the output format specified through the WRFMT bits is RGB. Ordered dither is available only for 18bpp. So, when ODE bit is equal to 1, set WRFMT at 18bpp format. When ODE bit is equal to 0, WPF dither method is specified by DITH[1:0] in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.
21 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18 to 16	ROT[2:0]	All 0	R/W	Rotation Processing Select These bits select the flipping processing to be applied to the WPFn output image. **Figure 33.18** shows the correspondence between the original image and the flipping result according to each setting. 0: No flipping 1: Vertical flipping Note that the destination address setting should be changed according to the setting of these bits. For details, refer to **Section 33.3.2.7(9)**. When the LIF module is used (VI6_LIF_CTRL.LIF_EN = 1), set 0 to ROT[2:0].

Bit	Bit Name	Initial Value	R/W	Description
15	SPYCS	0	R/W	WPF Output Mode Setting 1 When the output format is YUY2, set this bit to 1 and set the WRFMT bits to 71 (H’47). When the output format is YVYU, set this bit and the SPUVS bit to 1 and set the WRFMT bits to 71 (H’47). In other cases, set this bit to 0.
14	SPUVS	0	R/W	WPF Output Mode Setting 2 When the output format is NV61, set this bit to 1 and set the WRFMT bits to 65 (H’41). When the output format is NV21, set this bit to 1 and set the WRFMT bits to 66 (H’42). When the output format is YVYU, set this bit and the SPYCS bit to 1 and set the WRFMT bits to 71 (H’47). In other cases, set this bit to 0.
13, 12	DITH[1:0]	All 0	R/W	Ordered Dither (mode B) Enable/Disable When the output format specified through the WRFMT bits is RGB with 18 bpp (260000 colors) or less, the color reduction processing is applied to match the number of colors. The color reduction processing may generate the artifacts of pseudo gradation, which can be suppressed through dithering. The DITH bits enable or disable dithering during color reduction. When the output format specified through the WRFMT bits is YCbCr, specify 0 in these bits. And when VI6_WPFn_OUTFMT.CSC is set to 1, specify 0 in these bits even in the case that the output format specified through the WRFMT bits is RGB. 0: Dithering (mode B) is disabled 3: Dithering (mode B) is enabled 1, 2: Setting prohibited When ODE bit in the register is 1, set this bit to 0.
11 to 9	WRTM[2:0]	All 0	R/W	CSC Conversion Expression Setting These bits select the expression for color space conversion. The conversion direction is RGB to YCbCr when the format specified in the WRFMT bits is RGB, or YCbCr to RGB when the format is YCbCr. 0: BT.601 YCbCr [16,235/240] ↔ RGB [0,255] 1: BT.601 YCbCr [0,255] ↔ RGB [0,255] 2: BT.709 YCbCr [16,235/240] ↔ RGB [0,255] 3: BT.709 YCbCr [16,235/240] ↔ RGB [16,235] 4 to 7: Setting prohibited
8	CSC	0	R/W	Color Space Conversion Setting Enables or disables YCbCr ↔ RGB color space conversion to be executed in the WPFn. The characteristics of color space conversion are determined by the WRTM setting. There are some points to be noted about the relationship between the CSC setting and output format (WRFMT). For details refer to (*1) in **Section 33.3.2.6(3)**, RPFn Input Format Registers (VI6_RPFn_INFMT). 0: Color space is not converted. 1: Color space is converted.
7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
6 to 0	WRFMT[6:0]	All 0	R/W	WPF Output Image Format Setting These bits select the format of the image output from the WPFn to the external memory from among those listed in **Table 33.20** and **Table 33.21**. [Note 1] Number of output pixels When YCbCr4:2:2 is specified through WRFMT, the horizontal size of the output image should be a multiple of 2 pixels. When YCbCr4:2:0 is specified, the vertical and horizontal sizes of the output image should be multiples of 2 pixels. Specify an appropriate data flow of the source RPF -> DPR -> target WPF so that the size of the image input to the target WPF satisfies the above restrictions. In particular, when the data flow includes a module or a function that modifies (up-scales, down-scales, or clips) the image size, take special care about the module or function settings. [Note 2] Output lines in YCbCr4:2:0 In the YCbCr4:2:0 output format, the number of chrominance lines in the vertical direction is one-half the number of luminance lines. For this reason, the WPF outputs only even-numbered chrominance lines (lines 0, 2, 4, 6, ...) (conversion from (A) to (B) in. When vertical flipping is also specified through the ROT bits, the flipping processing is executed last and the chrominance line locations are inverted (lines 1, 3, 5, 7, …) in the output image ((C) in **Figure 33.19**). [Note 3] Down sampling of CbCr in horizontal direction in YCbCr4:2:0 or YCbCr4:2:2 In the YCbCr4:2:2 or YCbCr4:2:0 output format, method of down sample of Cb/Cr in horizontal direction is average of neighbor two pixels.

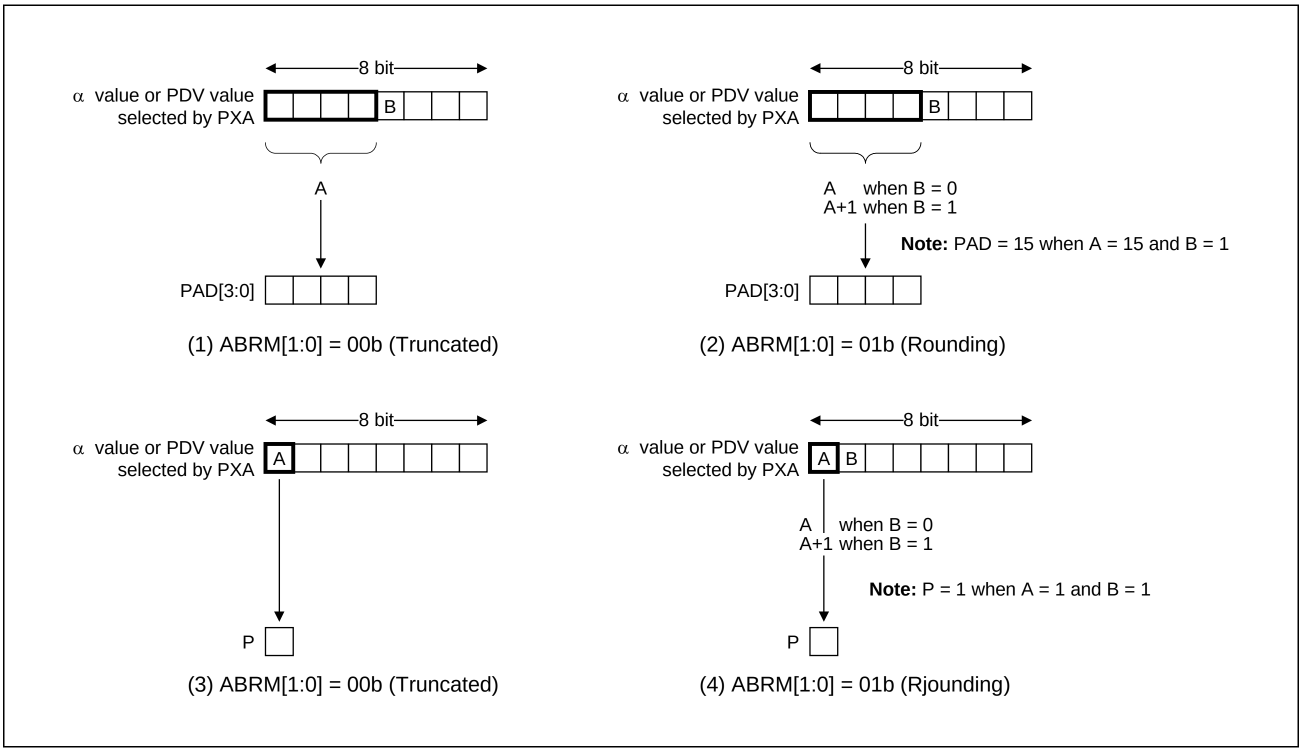


Figure 33.17 Selection of PAD Value and Reduction of Bit Count through PXA Setting

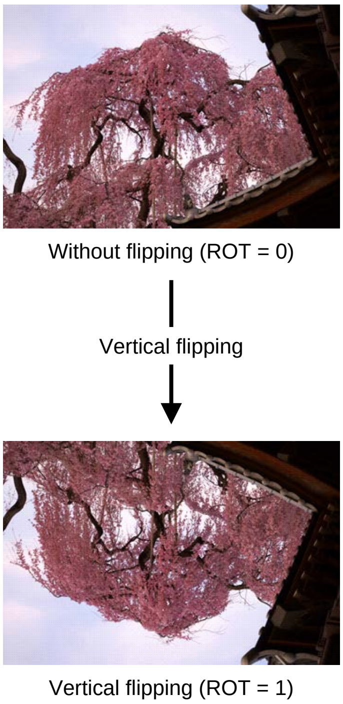


Figure 33.18 Correspondence between Original Image and Flipping Result according to ROT Setting

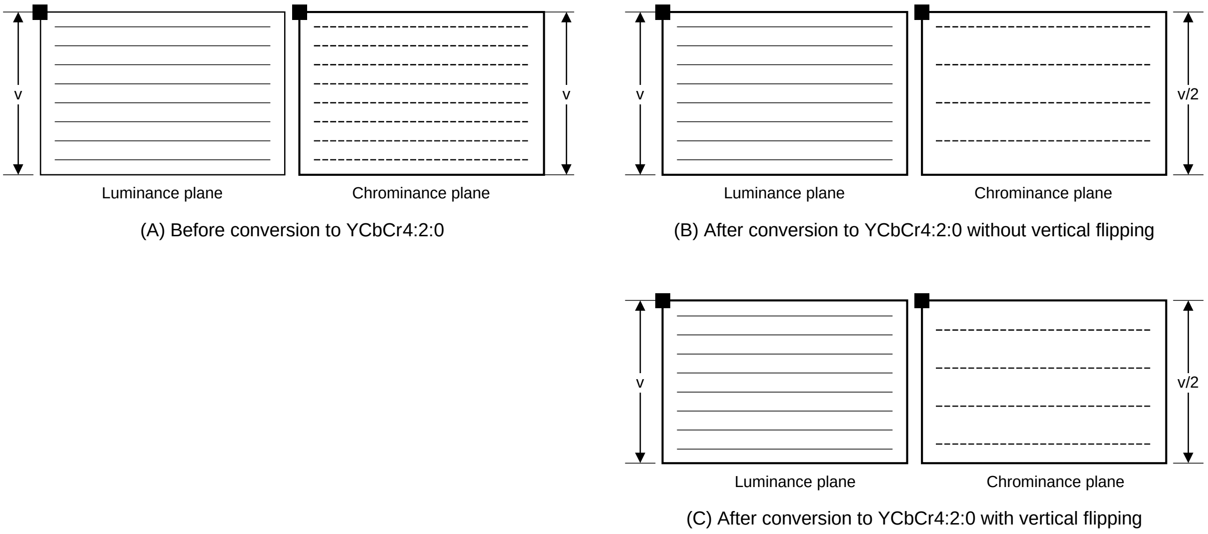


Figure 33.19 Chrominance Output Lines in YCbCr4:2:0 and Vertical Flipping Result

Table 33.20 Packed RGB Formats for WPF Output

			upper row - address / bottom row - bit field																															
			n								n+1								n+2								n+3							
WRFMT[6:0]	Bit per pixel	Phase	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
H'00	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
H'01	12	—	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
H'02			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	0	0	0	0
H'03	—	—	Reserved								Reserved								Reserved								Reserved							
H'04	15	—	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'05			R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	0
H'06	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'07	18	—	PAD								0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
H'08			R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	PAD							
H'09			0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	PAD							
H'0A			PAD								R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0
H'0B			PAD								0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0
H'0C			0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	PAD							
H'0D			PAD								R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0
H'0E			R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	PAD							
H'0F		0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1
		1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2
		2	G2	G2	B2	B2	B2	B2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3
H'10		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1	R1	R1	R1	R1	G1	G1
		1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
		2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	0	0	0	0	0	0
H'11		0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
H'12		0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1	0	0
		1	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2	0	0
		2	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3	0	0
H'13	24	—	PAD								R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
H'14			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	PAD							
H'15		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
H'16	18	—	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	0	0	0	0	0	0	0	G0	G0	G0	B0	B0	B0	B0	B0	B0
H'17			0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
H'18	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
H'19	12	—	PAD				R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
H'1A		—	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	PAD			
H'1B	15	—	P	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'1C		—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	P
H'1D	12	—	PAD				B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1
H'1E		—	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1	PAD			
H'1F	15	—	P	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1
H'20		—	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	P
H'21	18	0	0	0	B0	B0	B0	B0	B0	B0	0	0	G0	G0	G0	G0	G0	G0	0	0	R0	R0	R0	R0	R0	R0	0	0	B1	B1	B1	B1	B1	B1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	R1	R1	R1	R1	R1	R1	0	0	B2	B2	B2	B2	B2	B2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	R2	R2	R2	R2	R2	R2	0	0	B3	B3	B3	B3	B3	B3	0	0	G3	G3	G3	G3	G3	G3	0	0	R3	R3	R3	R3	R3	R3
H'22	24	—	PAD								B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
H'23	16	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0
H'24 to H'3F	—	—	Reserved								Reserved								Reserved								Reserved							

Table 33.21 Packed YCbCr Formats for WPF Output

WRFMT[6:0]	Packed YCbCr Output Format	Reference
H'40	YCbCr4:4:4 semi-planar	**Figure 33.9***4
H'41	YCbCr4:2:2 semi-planar (NV16, NV61*1)	
H'42	YCbCr4:2:0 semi-planar (NV12, NV21*1)	
H'43 to H'45	Reserved	—
H'46	YCbCr4:4:4 interleaved	**Figure 33.10***4
H'47	YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3)	
H'48	YCbCr4:2:2 interleaved type 1	
H'49	YCbCr4:2:0 interleaved *5	
H'4A	YCbCr4:4:4 planar	**Figure 33.11***4
H'4B	YCbCr4:2:2 planar (YV16)	
H'4C	YCbCr4:2:0 planar (YV12, I420)	
H'4D to H'7F	Reserved	—

Note 1. When the output format is NV61 or NV21, set SPUVS (bit 14) to 1.

Note 2. When the output format is YUY2, set SPYCS (bit 15) to 1.

Note 3. When the output format is YVYU, set SPUVS (bit 14) to 1 and SPYCS (bit 15) to 1.

Note 4. **Figure 33.12** shows the definition of memory address for each pixel in **Figure 33.9** to **Figure 33.11**.

Note 5. Each line of plane is written twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).

For details of each YCbCr format, refer to **Figure 33.9**, **Figure 33.10**, **Figure 33.11**, **Figure 33.12**. In these figures, registers for the RPF are indicated; read them as registers for the WPF as follows.

(RPF registers in the figures)	→	(Corresponding WPF registers)
VI6_RPFn_SRCM_ADDR_Y.SRCM_ADDR_Y	→	VI6_WPFn_DSTM_ADDR_Y.DSTM_ADDR_Y
VI6_RPFn_SRCM_ADDR_C0.SRCM_ADDR_C0	→	VI6_WPFn_DSTM_ADDR_C0.DSTM_ADDR_C0
VI6_RPFn_SRCM_ADDR_C1.SRCM_ADDR_C1	→	VI6_WPFn_DSTM_ADDR_C1.DSTM_ADDR_C1

(5) WPFn Data Swapping Registers (VI6_WPFn_DSWAP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	P_LLS	P_LWS	P_WDS	P_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	P_LLS	0	R/W	WPF Output Data Swapping in LONG LWORD Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 33.15**. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled
2	P_LWS	0	R/W	WPF Output Data Swapping in long word Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 33.15**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled
1	P_WDS	0	R/W	WPF Output Data Swapping in Word Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 33.15**. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled
0	P_BTS	0	R/W	WPF Output Data Swapping in Byte Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 33.15**. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled

Table 33.15 shows the data order before and after swapping according to the long long word, long word, word, and byte swapping settings.

When data order in memory for each format is the same as in **Table 33.20** for RGB format and **Figure 33.9** to **Figure 33.12** for YCbCr format, set 1111b to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is not the same as the definition, change data order within 16byte unit by these bits as shown in **Table 33.15**.

(6) WPFn Rounding Control Registers (VI6_WPFn_RNDCTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CBRM	—	—	ABRM[1:0]		ATHRESH[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CLMD[1:0]		—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	CBRM	0	R/W	Bit Count Reduction Method Selection for Data Storage in Packed RGB This bit specifies the method for reducing the number of bits when data is stored in the bit fields indicated as R, G, and B in **Table 33.20** and the target bit fields are not 8 bits. 0: Bit count conversion: The lower-order bits are truncated 1: Bit count conversion: Rounding (rounding off)
27, 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25, 24	ABRM[1:0]	All 0	R/W	Bit Count Reduction Method Selection for Data Storage in PAD These bits specify the method for reducing the number of bits when the data selected through VI6_WPFn_OUTFMT.PXA is stored in the bit fields indicated as PAD or P in **Table 33.20** and the target bit field is four bits or one bit. A value of 10b can be specified only when the packed RGB format specified through VI6_WPFn_OUTFMT.WRFMT includes a 1-bit P field. In this case, when the data selected through VI6_WPFn_OUTFMT.PXA is greater than the ATHRESH value, 1 is stored in the P field; when the selected data is not greater than the ATHRESH value, 0 is stored. 00b: Bit count conversion: The lower-order bits are truncated 01b: Bit count conversion: Rounding (rounding off) 10b: Bit count conversion: Comparison with the threshold value (this setting is allowed only when the storage field is one bit) 11b: Setting prohibited
23 to 16	ATHRESH [7:0]	All 0	R/W	Threshold for Conversion to 1-Bit α Data These bits specify the threshold value used for conversion from 8-bit α data to one bit when the ABRM bits are set to 10b. When the 8-bit α value before bit count reduction is equal to or smaller than the ATHRESH value, 0 is stored as the reduced 1-bit α data. In other cases, 1 is stored as the 1-bit α data. A value from 0 to 255 can be specified.
15, 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13, 12	CLMD[1:0]	All 0	R/W	Color Data Clipping These bits specify the method for clipping the YCbCr color data output from the WPF. When RGB color data is output from the WPF, specify 0 in these bits. 00b: Output value is not clipped (0-255) 01b: Output value is clipped: YCbCr mode 1 (16-235 (Y), 16-240 (Cb/Cr)) 10b: Output value is clipped: YCbCr mode 2 (Y/Cb/Cr = 1-254) 11b: Setting prohibited
11 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

(7) WPFn Destination Y Plane Memory Stride Registers (VI6_WPFn_DSTM_STRIDE_Y)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PICT_STRD_Y[15:0]	All 0	R/W	Memory Stride of Destination Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the destination picture in the external memory to be written to by the WPFn as shown in **Figure 33.20**. A value from H'0000 to H'FFFF can be specified.

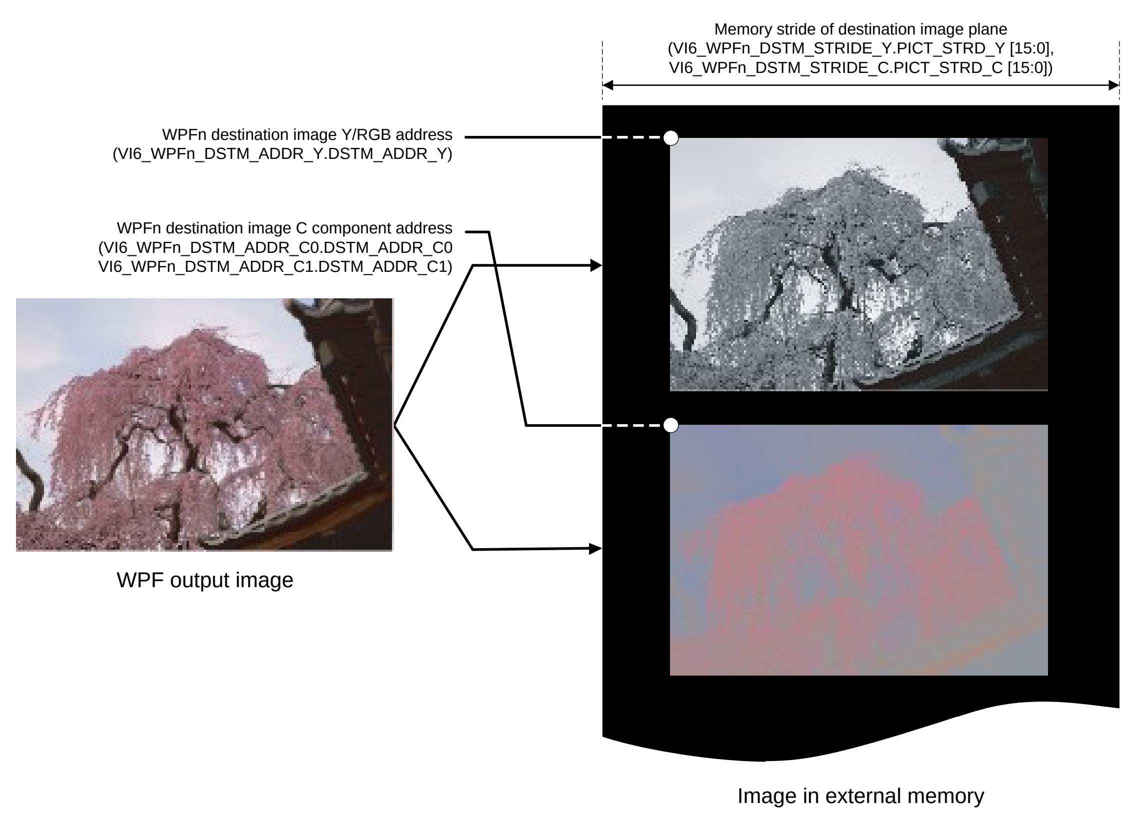


Figure 33.20 Writing Image Data to Destination Area in WPFn

(8) WPFn Destination C Plane Memory Stride Registers (VI6_WPFn_DSTM_STRIDE_C)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	PICT_STRD_Y[15:0]	All 0	R/W	Memory Stride of Destination Picture C Plane These bits specify in 1-byte units the memory stride for the C plane of the destination picture in the external memory to be written to by the WPFn as shown in **Figure 33.20**. When the WPFn outputs images in an RGB format, this setting is not used. When the WPFn outputs images in YCbCr planar format, this setting is applied to both the Cb and Cr planes. A value from H'0000 to H'FFFF can be specified.

(9) WPFn Destination Y/RGB Address Registers (VI6_WPFn_DSTM_ADDR_Y)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_Y[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_Y[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_Y[31:0]	All 0	R/W	Destination Image Y/RGB Plane Storing Address These bits specify in 1-byte units the address for storing the destination-image Y plane or packed RGB plane to be written to by the WPFn in the method described later. A value from H'0000 0000 to H'FFFF FFFF can be specified.

[Destination Address Specification Method]

When flipping is not used, the start address of a frame (address FHA shown in **Figure 33.21**) should be specified as the destination address. When flipping is used, the destination address is not the frame start address (FHA); one of addresses A0 and A2 shown in **Figure 33.21** should be selected according to the combination of desired flipping (VI6_WPFn_OUTFMT.ROT setting).

To strictly define locations A0 and A2, let the horizontal size of the output image be H, the vertical size of the output image be V, and the memory stride (VI6_WPFn_DSTM_STRIDE_Y/C setting) be S as shown in **Figure 33.21**. Calculate the destination address (one of A0 and A2) using the formula shown in **Table 33.22** and specify it in the destination address storing register.

The values of variables L in **Table 33.22** depend on the other register settings and luminance and chrominance components. These values should be obtained by referring to **Table 33.23** when calculating the address to be specified in VI6_WPFn_DSTM_ADDR_Y, or **Table 33.24** when calculating the address to be specified in VI6_WPFn_DSTM_ADDR_C0 or VI6_WPFn_DSTM_ADDR_C1.

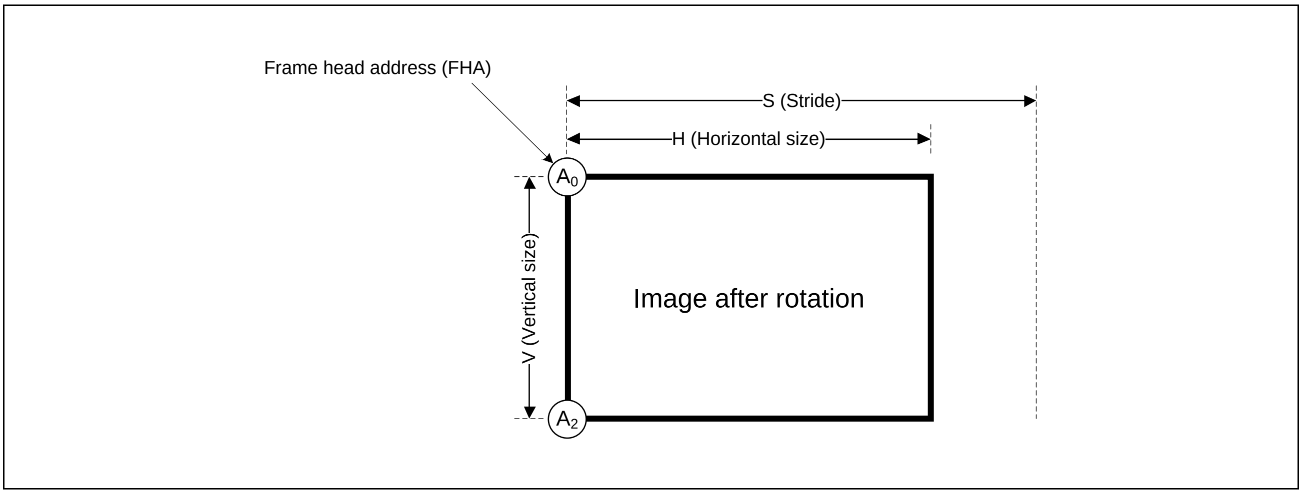


Figure 33.21 Location of Destination Address to be Specified for Flipping

Table 33.22 Destination Address A_0, A_1, A_2, and A3 Calculation Formulas

VI6_WPFn_OUTFMT.ROT Setting	Formula for Calculating Address to be Set in VI6_WPFn_DSTM_ADDR_Y, VI6_WPFn_DSTM_ADDR_C0, and VI6_WPFn_DSTM_ADDR_C1
0	A_0 = FHA
1	A_2 = FHA + (V×L – 1) × S

Table 33.23 Value of L according to VI6_WPFn_OUTFMT.WRFMT Setting (for RGB and Luminance Y Address Calculation)

VI6_WPFn_OUTFMT.WRFMT	L
73	0.5
0 to 2, 4 to 35, 64 to 66, 70 to 72, or 74 to 76	1

Table 33.24 Value of L according to VI6_WPFn_OUTFMT.WRFMT Setting (for Chrominance C0 and C1 Address Calculation)

VI6_WPFn_OUTFMT.WRFMT	L
70 to 73	Not defined
66, 76	0.5
64 to 65, 74 to 75	1

(10) WPFn Destination Chroma Address Registers 0 (VI6_WPFn_DSTM_ADDR_C0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_C0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_C0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_C0[31:0]	All 0	R/W	Destination Image C Plane Storing Address 0 These bits specify in 1-byte units the address for storing the destination-image C plane to be written to by the WPFn. Refer to the description of VI6_WPFn_DSTM_ADDR_Y for settings. Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in **Table 33.21** or the Cb plane when a planar format is selected. When an interleaved format is selected or the output is in an RGB format, this setting is not used. A value from H’0000 0000 to H’FFFF FFFF can be specified. Refer to **Figure 33.21** in **Section 33.3.2.7(10)** for settings.

(11) WPFn Destination Chroma Address Registers 1 (VI6_WPFn_DSTM_ADDR_C1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_C1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_C1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_C1[31:0]	All 0	R/W	Destination Image C Plane Storing Address 1 These bits specify in 1-byte units the address for storing the Cr plane when the WPFn outputs images to the external memory in YCbCr planar format shown in **Table 33.21**. Refer to the description of VI6_WPFn_DSTM_ADDR_Y for settings. This setting is not used when the WPF outputs in YCbCr format that is not a planar format or in an RGB format. A value from H’0000 0000 to H’FFFF FFFF can be specified. Refer to **Figure 33.21** in **Section 33.3.2.7(10)** for settings.

(12) WPFn LIF Write Back Control Registers (VI6_WPFn_WRBCK_CTRL: n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WBMD[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	WBMD[1:0]	All 0	R/W	Display Data Write Back Control This bit is used for selecting the write back mode when the value of VI6_LIFn_CTRL.LIF_EN bit is set to 1. 0: Write Back Disabled 1: Write Back Enabled

33.3.2.8 DPR Control Registers

(1) Concept of DPR Settings

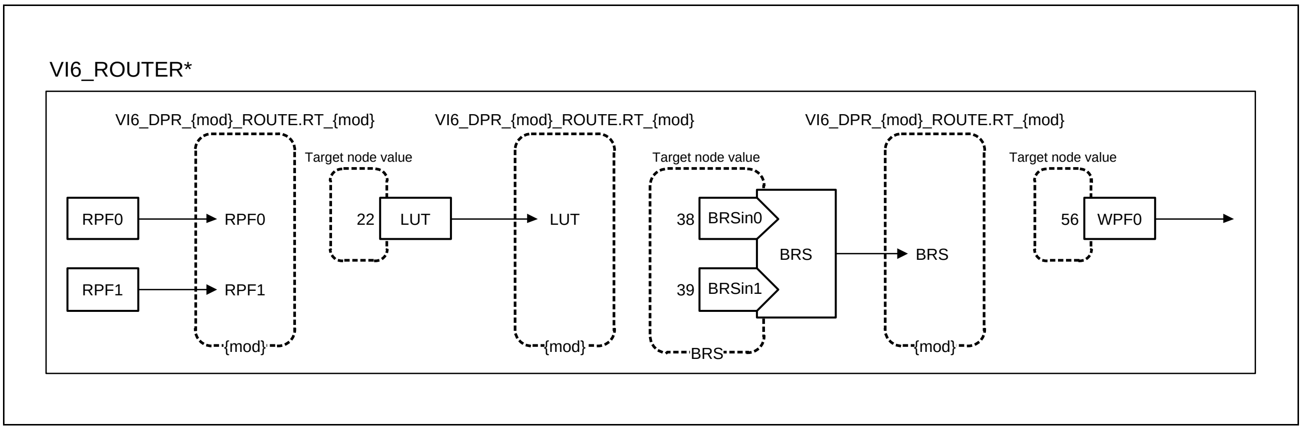


Figure 33.22 Node Register Names and Target Node Values on Data Path Router

In the VSPD internal data path, the order of processes can be specified as desired. The module for performing each process has a unique node value. Set each bit field in VI6_DPR_*_ROUTE to an appropriate node value shown in **Figure 33.22** to specify the target node to be connected behind each module.

For DPR settings, all of the following restrictions should be observed. If any of them is violated, even the WPF paths operating correctly at that time will be affected as well as the WPF paths connected through the DPR, and correct operation will not be guaranteed.

1. Specify 63 for the output node values of all RPFs and processing modules that are not used in the DPR. Here, make sure that no module is connected to a module for which 63 is specified as the node value. Note: VI6_DPR_{mod}_ROUTE.RT are not implemented in VSPD which doesn't have {mod}. See **Figure 33.1** whether VSPD has {mod} or not.
2. When specifying a value other than 63 for an output node value in the DPR, make sure that valid inputs (RPF0 to RPF1 or virtual RPF) and a target WPF are determined.
3. Only one module can be connected to each module; specifying the same target node value for two or more modules is prohibited.
4. Desired modules can be connected between each RPF and BRS input port, but all RPFs specified as the sources for a BRS input port should have the same target WPF.
5. Make appropriate routing or RPF register settings so that the color space formats (RGB/YCbCr) for all BRS input ports are the same.
6. Do not connect the output of any module as the input to the same module (in the BRS case, any input port) even when there is another module between the output and input (creating a loop is prohibited).
7. Each node can be used only once throughout all paths from RPFn to WPFn. When a module shown in **Figure 33.22** is assigned in one RPF to WPF path, it cannot be used in another RPF to WPF path.
8. While a WPF is operating, modifying the DPR connection settings in VI6_DPR_*_ROUTE is prohibited for modules used by the WPF but allowed for modules not used by the WPF. Be careful not to accidentally modify the settings of the modules included in the WPF path that is operating.
9. Be careful to do the following setting when BRS is used.
   VI6_DPR_ILV_BRS_ROUTE.BRSSEL = 1

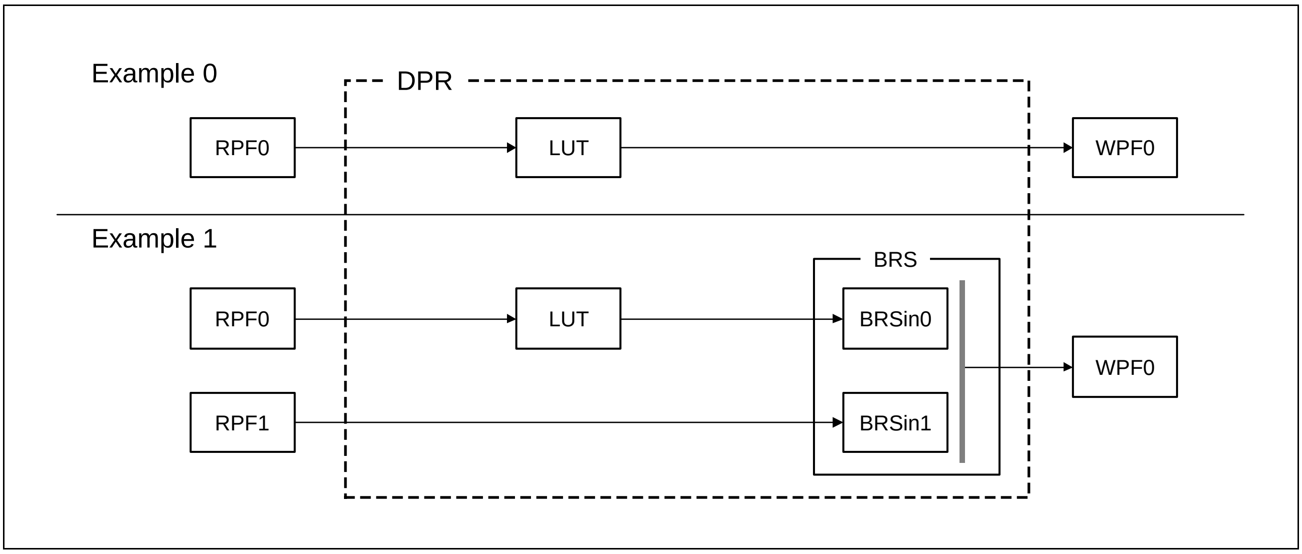


Figure 33.23 Examples of Internal Data Path Routing

Figure 33.23 shows examples of internal data path routing. Example 0 is WPF0 processing (RPF0 is the source RPF), and example 1 is also WPF0 (RPF0 to RPF1 are the source RPFs). Each example has the configuration shown in **Figure 33.23**. Example 0 performs LUT (e.g. γ correction) processing. Example 1 performs LUT processing (e.g. γ correction), for input 0 (RPF0) and then applies blending or raster operation between the resultant data and input data 1 (RPF1). The VI6_ DPR_*_ROUTE settings for these examples are shown in **Table 33.25**. The bit fields for the modules that are not used in the examples should be set to 63.

Table 33.25 VI6_DPR_*_ROUTE Register Settings in Connection Examples Shown in **Figure 33.23**. Although both examples show in **Figure 33.23** include operation in image processing modules, connect the RPF to the WPF directly when only image format conversion or packed format conversion is required.

Table 33.25 Examples of Internal Data Path Routing

	Register Name	Setting	
Example 0	VI6_DPR_RPF0_ROUTE.RT_RPF0	22	(To LUT)
	VI6_DPR_RPF1_ROUTE.RT_RPF1	63	(UNUSED)
	VI6_DPR_LUT_ROUTE.RT	56	(To WPF0)
	VI6_DPR_ILV_BRS_ROUTE.RT	63	(UNUSED)
Example 1	VI6_DPR_RPF0_ROUTE.RT_RPF0	22	(To LUT)
	VI6_DPR_RPF1_ROUTE.RT_RPF1	39	(To BRSin1)
	VI6_DPR_LUT_ROUTE.RT	38	(To BRSin0)
	VI6_DPR_ILV_BRS_ROUTE.RT	56	(To WPF0)

(2) RPFn Routing Register (VI6_DPR_RPFn_ROUTE : n = 0, 1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	RT_RPFn[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	RT_RPFn [5:0]	All 0	R/W	RPFn Target Node Value These bits specify the target node value for RPFn. When using RPFn, refer to **Figure 33.22** for settings. When RPFn is not started through the VI6_WPFn_SRCRPF setting, specify 63.

(3) WPFn Timing Control Register (VI6_DPR_WPFn_FPORCH : n = 0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	FP_WPFn[5:0]						—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 8	FP_WPFn [5:0]	All 0	R/W	WPFn Internal Operation Timing Setting Specify 5.
7 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

(4) {mod} Routing Register (VI6_DPR_{mod}_ROUTE : {mod}=ILV_BRS, LUT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	BRSSEL	—	—	—	—	FXA[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	FP[5:0]						—	—	RT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	BRSSEL	0	R/W	This bit is valid only for VI6_DPR_ILV_BRS_ROUTE. Always specify 1.
27 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	FXA[7:0]	All 0	R/W	Fixed α Output Value for {mod} The {mod} does not support input/output of the α value. The α value input to the {LUT} is discarded, and the fixed α value specified in these bits is always output from the {LUT}. A value from 0 to 255 can be specified. These bits are valid for LUT module. For BRS module, these bits are reserved.
15, 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 8	FP[5:0]	All 0	R/W	{mod} Internal Operation Timing Setting Specify 0.
7, 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	RT[5:0]	All 0	R/W	{mod}Target Node Value These bits specify the target node value for the {mod}. When using the {mod}, refer to **Figure 33.22** for settings. When not using the {mod}, specify 63.

33.3.2.9 LUT Control Register

(1) LUT Control Register (VI6_LUT_CTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LUT_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	LUT_EN	0	R/W	1D-LUT Enable/Disable Enables or disables the 1D-LUT function by the LUT. When the 1D-LUT is used, the color component curve information needs to be set separately in the LUT table. For the LUT table settings, refer to **Section 33.4.7.1**. 0: 1D-LUT function is disabled 1: 1D-LUT function is enabled

In the LUT, various image processing, such as curves with high operation load (e.g., γ correction), negative-positive conversion, and gain adjustment of images, can be achieved by the data replacement processing by the 1D-LUT. As shown in **Figure 33.24**, the LUT replaces each component of the input pixel data using the set replacement table of 256 entries. For example, if the LUT is set as in **Figure 33.25**, when there is an input of 150, the data stored in address 150 of the 1D-LUT is read and output as the LUT output. **Figure 33.25** shows a case in which the input and output become equal for convenience in explaining.

In the LUT settings shown in **Figure 33.26**, the input bits are reversed. This has the effect of negative-positive flipping. In **Figure 33.27**, γ correction (γ = 1.8 is shown as an example) is possible. As described above, information to be set in the LUT indicates LUT processing characteristics. If the same value is set for each component in the LUT, an equal effect can be obtained for each component of the input image when it is processed. If the LUT is set with different characteristics for each component, the processing characteristics can be changed for each component.

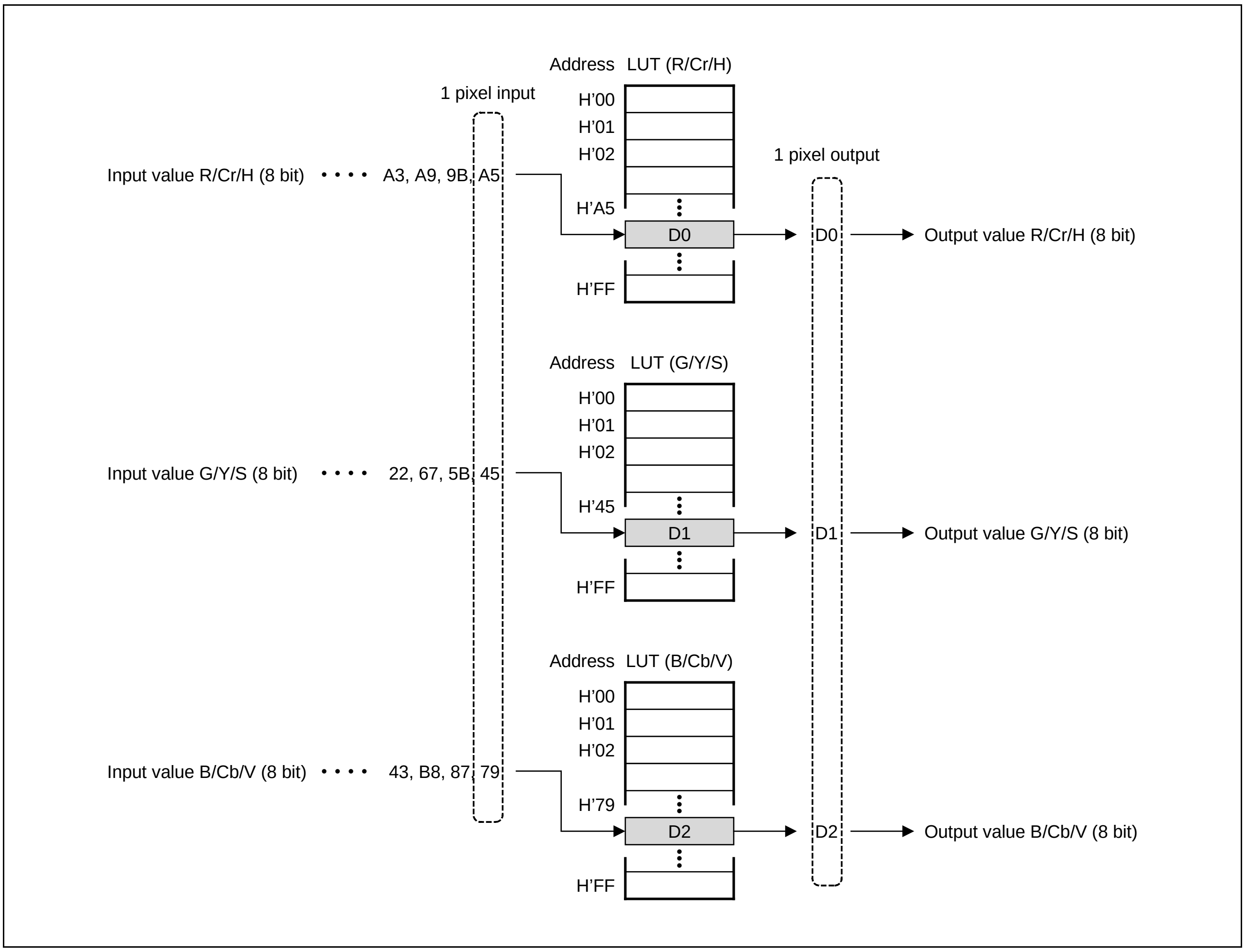


Figure 33.24 Relationship between Input and Output for 1D-LUT Table

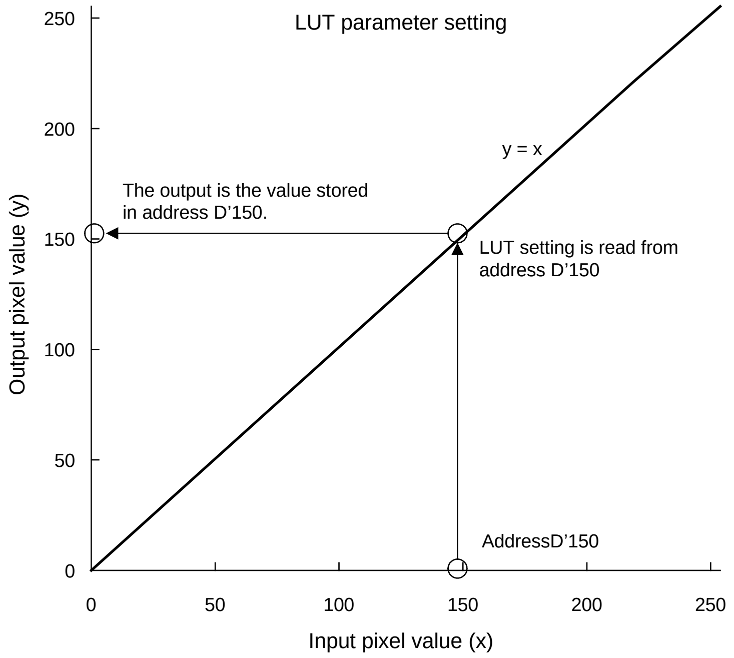


Figure 33.25 Setting Example in Which Output Becomes Equal to Input

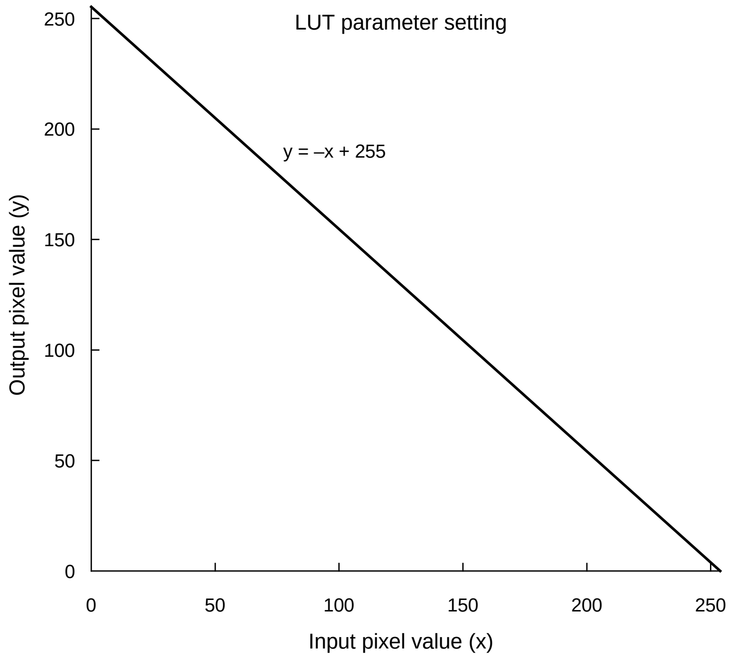


Figure 33.26 Setting Example of Negative-Positive Conversion

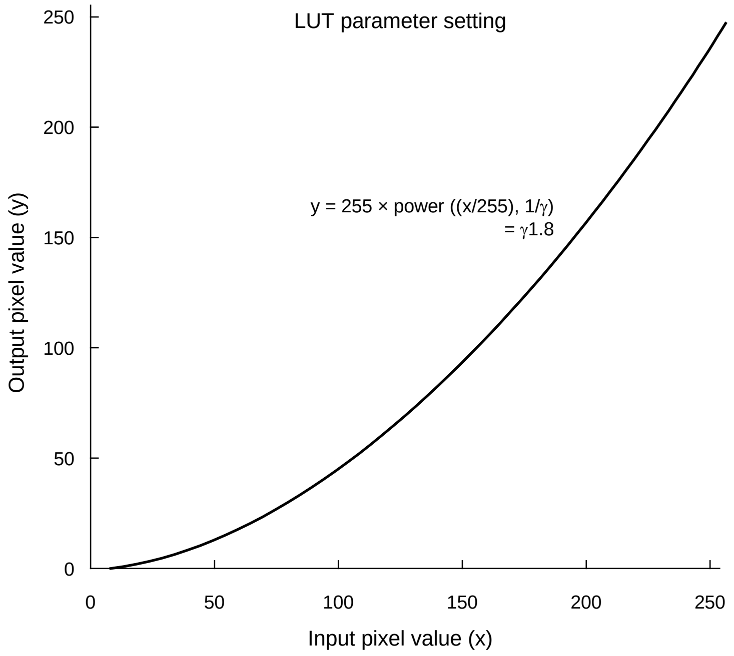


Figure 33.27 Setting Example of γ Correction

33.3.2.10 BRS Control Registers

(1) BRS Input Control Register (VI6_BRS_INCTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	NRM	—	—	—	—	—	—	—	—	—	—	D1ON	D0ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ODE1	DITH1[2:0]			ODE0	DITH0 [2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	NRM	0	R/W	Color Data Normalization Enables or disables division by the α value of the color data in BRS blending operation. This is used when converting the RGB color data format to which the α value is multiplied (pre-multiplied color) into the RGB color data format to which the α value is not multiplied (non pre-multiplied color). Do not use this for the YCbCr format. 0: Divider (DIV unit in **Figure 33.28**) does not divide the color value by α 1: Divider (DIV unit in **Figure 33.28**) divides the color value by α
27 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	D1ON	0	R/W	Ordered dither (mode B) Enable of BRS Input 1 Enables or disables dithering (color reduction) of BRS input 1 (BRSin1 in **Figure 33.28**). 0: Dithering (mode B) of BRSin1 is disabled 1: Dithering (mode B) of BRSin1 is enabled When ODE1 in this bit is set to 1, set 0 to this bit.
16	D0ON	0	R/W	Ordered dither (mode B) Enable of BRS Input 0 Enables or disables dithering (color reduction) of BRS input 0 (BRSin0 in **Figure 33.28**). 0: Dithering (mode B) of BRSin0 is disabled 1: Dithering (mode B) of BRSin0 is enabled When ODE0 in this bit is set to 1, set 0 to this bit.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	ODE1	0	R/W	Ordered Dither (mode A) of CH1 Input to BRS Enable/Disable 0: Ordered dither (mode A) of BRSin1 is disabled. 1: Ordered dither (mode A) of BRSin1 is enabled. Ordered dither is available only for 18bpp. So, when ODE1 bit is equal to 1, set DITH1 = 1. When ODE1 bit is equal to 0, BRSin1 dither method is specified by D1ON in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	DITH1[2:0]	All 0	R/W	Dithering of CH1 Input to BRS These bits specify the number of colors for pixels after dithering (color reduction) when dithering (color reduction) for pixel information is enabled through the D1ON bit. When dithering (color reduction) for pixel information is disabled, specify 0 in these bits. 0: Dithering of BRSin1 input image is disabled 1: Dithering of BRSin1 input image at 18 bpp (RGB666: 260000 colors) 2: Dithering of BRSin1 input image at 16 bpp (RGB565: 65535 colors) 3: Dithering of BRSin1 input image at 15 bpp (RGB555: 32768 colors) 4: Dithering of BRSin1 input image at 12 bpp (RGB444: 4096 colors) 5: Dithering of BRSin1 input image at 8 bpp (RGB332: 256 colors) 6, 7: Setting prohibited
3	ODE0	0	R/W	Ordered Dither (mode A) of CH0 Input to BRS Enable/Disable 0: Ordered dither (mode A) of BRSin0 is disabled. 1: Ordered dither (mode A) of BRSin0 is enabled. Ordered dither is available only for 18bpp. So, when ODE0 bit is equal to 1, set DITH0 = 1. When ODE0 bit is equal to 0, BRSin0 dither method is specified by D0ON in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.
2 to 0	DITH0 [2:0]	All 0	R/W	Dithering of CH0 Input to BRS These bits specify how to perform dithering of the CH0 input to the BRS. The setting method is the same as that for the DITH1 bits. Read the description of the DITH1 bits with BRSin0 and D0ON replacing BRSin1 and D1ON, respectively.

Figure 33.28 shows the configuration of the BRS. For the BRS inputs, there are two inputs from the DPR and one internal input as a virtual RPF. BRSin0 to BRSin1 are input ports that have the target node values shown in **Figure 33.22**, and they can be connected to any module on the DPR. The same color space (YCbCr or RGB) has to be used for the two inputs from the DPR to the BRS.

The virtual RPF inside the BRS is an input unit not connected to the DPR. It is called the “virtual RPF” because it outputs images internally created by the BRS. Starting of the virtual RPF is controlled by VI6_WPFn_SRCRPF.VIR_ACT2, and the single-color data created at the virtual RPF can be used for blending or raster operation (ROP) with data from the other input units BRSin0 to BRSin1. The color space for the single color to be set for the virtual RPF needs to match the color space of the two inputs from the DPR to the BRS. For this setting method, see **Section 33.3.2.10(4)**.

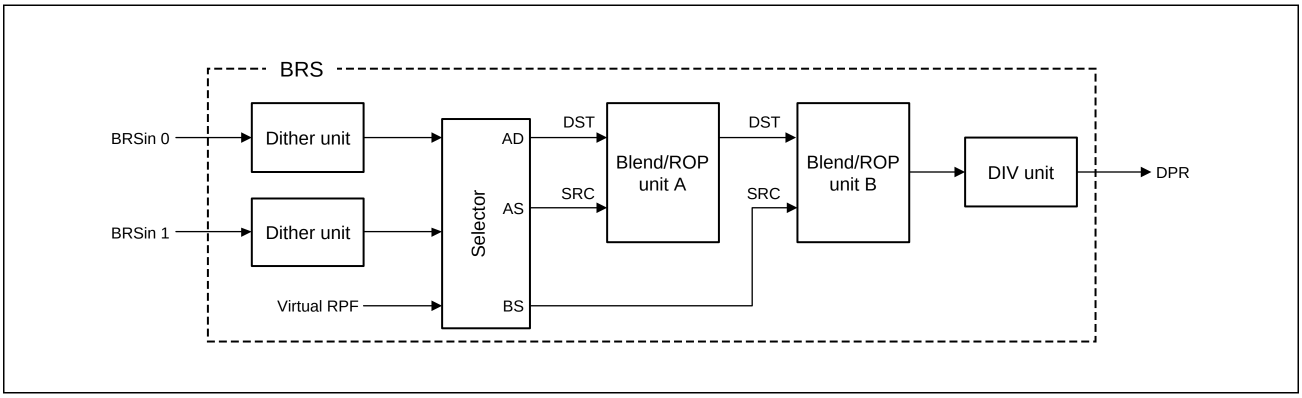


Figure 33.28 BRS Configuration

The selector in **Figure 33.28** is used to select the SRC and DST inputs to blending/ROP units A to B from BRSin0 to BRSin1 which are inputs from the DPR and the virtual RPF. The SRC and DST input sources for blending/ROP units A to B are either uniquely determined based on the configuration shown in **Figure 33.28** or selected as desired by registers. The input sources that can be arbitrarily selected by registers are AD, AS and BS, which correspond to the registers shown in **Table 33.26**.

Table 33.26 Correspondence between Selector Output Destinations and Register Bits for BRS

Selector Output	Output Destination	Register Bits
AD	Blending/ROP unit A - DST	VI6_BRSA_CTRL.DSTSEL
AS	Blending/ROP unit A - SRC	VI6_BRSA_CTRL.SRCSEL
BS	Blending/ROP unit B - SRC	VI6_BRSB_CTRL.SRCSEL

(2) Size Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_SIZE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VIR_HSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VIR_VSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	VIR_HSIZE [12:0]	All 0	R/W	Virtual RPF Horizontal Size These bits set the horizontal size of an image from the virtual RPF shown in **Figure 33.28**. A value from 1 to 1920 can be specified.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	VIR_VSIZE [12:0]	All 0	R/W	Virtual RPF Vertical Size These bits set the vertical size of an image from the virtual RPF shown in **Figure 33.28**. A value from 1 to 1080 can be specified.

The virtual RPF has only a function to output a fixed α value and a fixed pixel value. The virtual RPF can internally create a single-color image without accessing external memory via the MAU. Same as images from the other BRS input ports, a sublayer can be blended on an image created in this manner with the image used as the background (master layer). In turn, when using the image as a sublayer, it can be drawn on the master layer as a window.

(3) Display Location Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_LOC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	HCOORD [12:0]	All 0	R/W	Horizontal Coordinate of Virtual RPF Location on Master Layer These bits specify the horizontal coordinate of where to locate the left-edge pixel of the virtual RPF's layer, with the left-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 1919 can be specified. When the virtual RPF is specified as the master layer by VI6_WPFn_SRCRPF.VIR_ACT2, set these bits to 0.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	VCOORD [12:0]	All 0	R/W	Vertical Coordinate of Virtual RPF Location on Master Layer These bits specify the vertical coordinate of where to locate the top-edge pixel of the virtual RPF's layer, with the top-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 1079 can be specified. When the virtual RPF is specified as the master layer by VI6_WPFn_SRCRPF.VIR_ACT2, set these bits to 0.

(4) Color Information Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_COL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	COL_A[7:0]								COL_RCR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COL_GY[7:0]								COL_BCB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	COL_A[7:0]	All 0	R/W	Fixed α of Virtual RPF These bits set the fixed α value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.
23 to 16	COL_RCR [7:0]	All 0	R/W	Fixed R/Cr of Virtual RPF These bits set the fixed R/Cr value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.
15 to 8	COL_GY[7:0]	All 0	R/W	Fixed G/Y of Virtual RPF These bits set the fixed G/Y value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.
7 to 0	COL_BCB [7:0]	All 0	R/W	Fixed B/Cb of Virtual RPF These bits set the fixed B/Cb value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.

The transparency information and color information of the single color that is created by the virtual RPF are set in the bits of this register. As described earlier, the color information is set for the YCbCr or RGB color space. The color space to be set in this register depends on the register settings of the environment and other modules to which the BRS is connected by the DPR. Two cases can be considered. Since the α value (COL_A) is transparency information and irrelevant to the concept of color space, the same setting is made for either the YCbCr or RGB color space.

[Case 1: When an input other than the virtual RPF is used]

When the source RPF is connected to any one of the BRS input ports (BRSin0 to BRSin1) other than the virtual RPF and valid data is being supplied, the same color space data as the color space for the BRS inputs should be set in this register as the color space for the virtual RPF's color information. This is based on the restriction of “all BRS inputs must have the same color space”, as described in **Section 33.3.2.8(1)** or **Section 33.3.2.10(1)**.

[Case 2: When only the virtual RPF is used]

When only the virtual RPF is used as the source RPF of WPFn, RPFn is not connected to the BRS, as shown in **Figure 33.29**. Thus, there is no color space for another input that the color space for the virtual RPF has to follow, as in case 1. This means that the color space of the data output by the BRS is determined by the WPF setting.

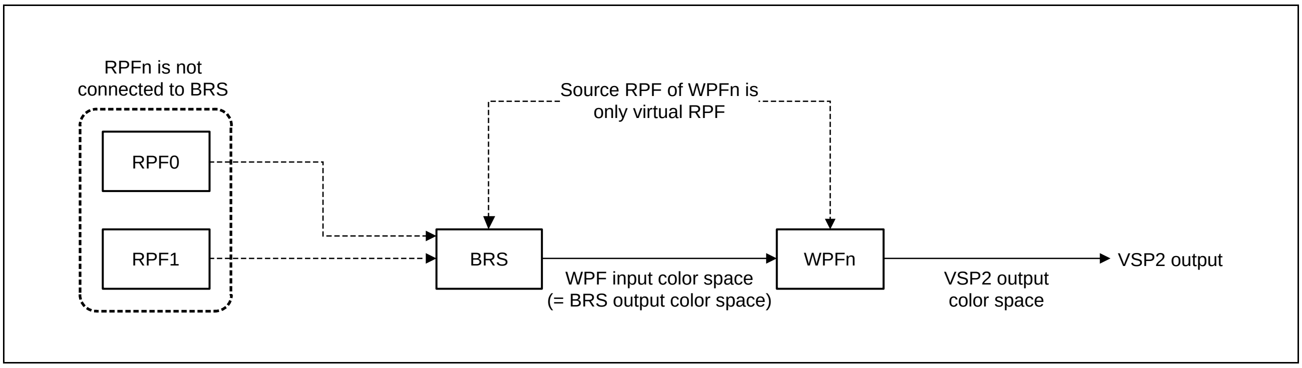


Figure 33.29 Relationship between DPR Connection and Color Space When Only Virtual RPF is Used

Figure 33.29, the output color space of the VSPD (= output color space of WPFn) is determined by VI6_WPFn_OUTFMT.WRFMT. When bit 6 in VI6_WPFn_OUTFMT.WRFMT (WRFMT [6]) is 0, the color space is RGB, while when it is 1, the color space is YCbCr. Next, the WPF input color space (= BRS output color space) is determined by the relationship between the WPF output color space and VI6_WPFn_OUTFMT.CSC. When VI6_WPFn_OUTFMT.CSC is 0, the WPF output color space and WPF input color space (= BRS output color space) are the same. When VI6_WPFn_OUTFMT.CSC is 1, the WPF output color space and WPF input color space (= BRS output color space) are the opposite. This relationship is summarized in **Table 33.27**.

The color space for the virtual RPF's color information should be set in this register according to the “BRS output color space” shown in **Table 33.27**.

Table 33.27 Relationship between WPF Output Color Space and BRS Output Color Space

VI6_WPFn_OUTFMT Register Bit Settings				BRS Output Color Space (= WPF Input Color Space)
Bit 6 in WRFMT		CSC		
0	(WPF output is RGB)	0	(YCbCr to RGB conversion is disabled)	RGB
0	(WPF output is RGB)	1	(YCbCr to RGB conversion is enabled)	YCbCr
1	(WPF output is YCbCr)	0	(RGB to YCbCr conversion is disabled)	YCbCr
1	(WPF output is YCbCr)	1	(RGB to YCbCr conversion is enabled)	RGB

(5) BRS Control Registers (VI6_BRSm_CTRL: m = A, B)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RBC	—	—	—	—	—	—	—	—	DSTSEL[2:0]			—	SRCSEL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	LBA2[11:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RBC	0	R/W	Operation Type of Blending/ROP Unit m (m = A, B) Specifies the operation type for blending/ROP unit m (m = A, B) shown in **Figure 33.28**. 0: ROP (raster operation) 1: Blending operation
30 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 20	DSTSEL[2:0]	All 0	R/W	Input Selection for DST Side of Blending/ROP Unit A These bits select the input for the DST side of blending/ROP unit A shown in **Figure 33.28**. These bits specify the connection between the BRS input port and the DST separately from the setting of connections between other modules and the BRS input port through the DPR. 000b: BRS input 0 (BRSin0) is input to DST 001b: BRS input 1 (BRSin1) is input to DST 100b: Virtual RPF is input to DST 010b, 011b, 101b to 111b: Setting prohibited [Note] The DSTSEL bits for blending/ROP unit B is reserved. The write value should always be 0.
19	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18 to 16	SRCSEL[2:0]	All 0	R/W	Input Selection for SRC Side of Blending/ROP Unit m (m = A, B) These bits select the input for the SRC side of blending/ROP unit m (m = A, B) shown in **Figure 33.28**.These bits specify the connection between the BRS input port and the SRC separately from the setting of connections between other modules and the BRS input port through the DPR. 000b: BRS input 0 (BRSin0) is input to SRC 001b: BRS input 1 (BRSin1) is input to SRC 100b: Virtual RPF is input to SRC 010b, 011b, 101b to 111b: Setting prohibited [Note] The SRCSEL bits for blending/ROP unit B are reserved. The write value should always be 0.
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 4	CROP[3:0]	All 0	R/W	Color Data ROP Operator These bits select the ROP operator of the color data in blending/ROP unit m (m = A, B). Select the opcode for ROP operation from **Table 33.28**.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	AROP[3:0]	All 0	R/W	α Data ROP Operator These bits select the ROP operator of the α data in blending/ROP unit m (m = A, B). Select the opcode for ROP operation from **Table 33.28**.

Table 33.28 ROP Operator of Blending/ROP Unit m (m = A, B)

Opcode	Operator
0000b	NOP(D)
0001b	AND(S & D)
0010b	AND_REVERSE(S & ~D)
0011b	COPY(S)
0100b	AND_INVERTED(~S & D)
0101b	CLEAR(0)
0110b	XOR(S ^ D)
0111b	OR(S | D)
1000b	NOR(~(S | D))
1001b	EQUIV(~(S ^ D))
1010b	INVERT(~D)
1011b	OR_REVERSE(S | ~D)
1100b	COPY_INVERTED(~S)
1101b	OR_INVERTED(~S | D)
1110b	NAND(~(S & D))
1111b	SET(all1)

(6) BRS Blend Control Registers (VI6_BRSm_BLD: m = A, B)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBES	CCMDX[2:0]			—	CCMDY[2:0]			ABES	ACMDX[2:0]			—	ACMDY[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COEFX[7:0]								COEFY[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CBES	0	R/W	Blending Expression Selection Selects the blending expression of the color data in the BRS (VI6_BRSm_CTRL.RBC = 1). Blending coefficients are specified by the CCMDX and CCMDY bits. 0: CCMDX * (DST color data) + CCMDY * (SRC color data) 1: CCMDX * (DST color data) - CCMDY * (SRC color data)
30 to 28	CCMDX[2:0]	All 0	R/W	Blending Coefficient X Selection These bits specify coefficient X used in the blending expression determined by the CBES bit. 000b: DST α data is used as blending coefficient X 001b: 255 - (DST α data) is used as blending coefficient X 010b: SRC α data is used as blending coefficient X 011b: 255 - (SRC α data) is used as blending coefficient X 100b: Fixed α value 0 (COEFX setting)
27	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26 to 24	CCMDY[2:0]	All 0	R/W	Blending Coefficient Y Selection These bits specify coefficient Y used in the blending expression determined by the CBES bit. 000b: DST α data is used as blending coefficient Y 001b: 255 - (DST α data) is used as blending coefficient Y 010b: SRC α data is used as blending coefficient Y 011b: 255 - (SRC α data) is used as blending coefficient Y 100b: Fixed α value 1 (COEFY setting) When the virtual RPF is specified as the master layer by VI6_WPFn_SRCRPF.VIR_ACT2, set these bits to 0.
23	ABES	0	R/W	Blending α Creation Expression Specifies the expression for creating α data after blending by blending/ROP unit m (m = A, B, C, D, E). α creation coefficients are specified by the ACMDX and ACMDY bits. 0: ACMDX * (DST α data) + ACMDY * (SRC α data) 1: ACMDX * (DST α data) - ACMDY * (SRC α data)
22 to 20	ACMDX[2:0]	All 0	R/W	α Creation Coefficient X These bits specify α creation coefficient X used in the α creation expression determined by the ABES bit. 000b: (α creation coefficient X) = (DST α data) 001b: (α creation coefficient X) = 255 - (DST α data) 010b: (α creation coefficient X) = (SRC α data) 011b: (α creation coefficient X) = 255 - (SRC α data) 100b: (α creation coefficient X) = Fixed α value 0 (COEFX setting) 101b to 111b: Setting prohibited
19	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	ACMDY[2:0]	All 0	R/W	α Creation Coefficient Y These bits specify α creation coefficient Y used in the α creation expression determined by the ABES bit. 000b: (α creation coefficient Y) = (DST α data) 001b: (α creation coefficient Y) = 255 - (DST α data) 010b: (α creation coefficient Y) = (SRC α data) 011b: (α creation coefficient Y) = 255 - (SRC α data) 100b: (α creation coefficient Y) = Fixed α value 1 (COEFY setting) 101b to 111b: Setting prohibited
15 to 8	COEFX[7:0]	All 0	R/W	Fixed α Value 0 These bits specify fixed α value 0 used when the CCMDX or ACMDX bits are set to 100b. A value from H’00 to H’FF can be specified.
7 to 0	COEFY[7:0]	All 0	R/W	Fixed α Value 1 These bits specify fixed α value 1 used when the CCMDY or ACMDY bits are set to 100b. A value from H’00 to H’FF can be specified.

33.3.2.11 LIF Control Registers

(1) LIF0 Control Register (VI6_LIF0_CTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	OBTH [11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PADL	—	—	—	CFMT	—	—	REQSEL	LIF_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	OBTH [11:0]	0	R/W	Buffer Threshold for Start Ready Notification to Display Module Always specify 1500, when LIF_EN is set to 1.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	PADL	0	R/W	Enable/Disable of padding with dummy lines to output-image to the DU. [LIF0 only] Enables or disables of padding with dummy lines to output-image to the DU. 0: Padding with dummy lines to output image to DU is disabled. 1: Padding with dummy lines to output image to DU is enabled.
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	CFMT	0	R/W	Chroma Format This bit selects the output format from the LIF module to the display module. When RGB format is used, this bit shall be set to 0. 0: YCbCr444 or RGB Format 1: YCbCr422 Format [Note] The DU cannot receive YCbCr422 format. Therefore, when CFMT is 1, YCbCr444 data which information contents is equal to YCbCr422 is output to DU as shown in **Figure 33.30**.
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	REQSEL	0	R/W	External Display Module Selection 0: This value is setting prohibited when 1 is set to LIF_EN 1: DU is selected as the destination external display module.
0	LIF_EN	0	R/W	Enable/Disable of Data Output to External Display Module Enables or disables data output from the LIF to the external display module (DU). 0: Data output to the external display module is disabled. 1: Data output to the external display module is enabled.

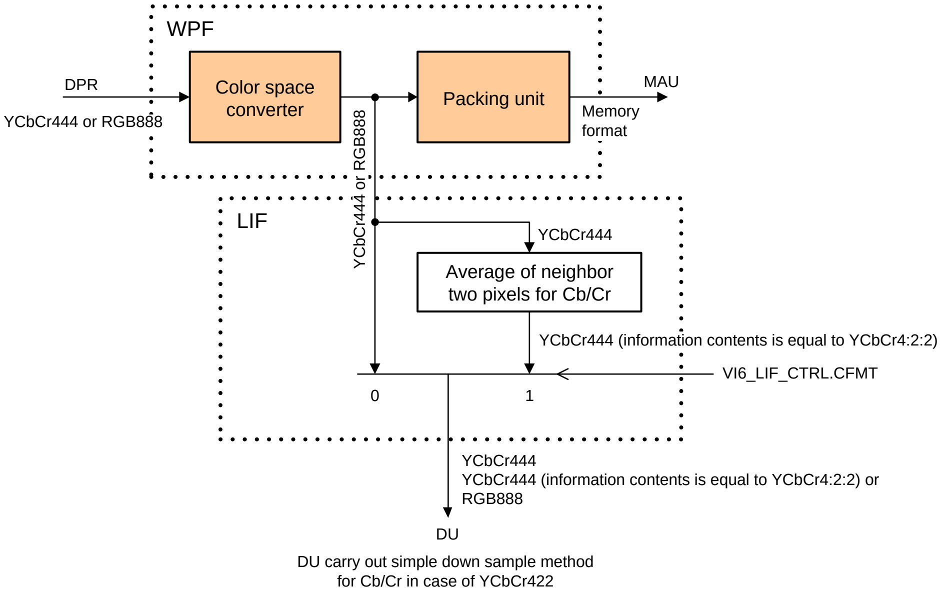


Figure 33.30 Data flow between LIF and DU

(2) LIF0 Clock Stop Buffer Control Register (VI6_LIF0_CSBTH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	HBTH[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LBTH[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	HBTH[11:0]	0	R/W	Buffer Threshold for Start Ready Notification to Display Module Always specify 1500, when LIF_EN is set to 1.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	LBTH[11:0]	0	R/W	Buffer Threshold for Clock Start in Dynamic Clock Control Set LBTH[11:0] = 0 (fixed value)

(3) LIF0 Buffer Attribute Register (VI6_LIF0_LBA)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LBA0	—	—	—	LBA1[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LBA2[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	LBA0	0	R/W	LIF Buffer Attribute Register0 Always specify 1, when LIF_EN is set to 1
30 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	LBA1[11:0]	0	R/W	LIF Buffer Attribute Register1 Always specify 1536, when LIF_EN is set to 1
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	LBA2[11:0]	0	R	LIF Buffer Attribute Register2 These bits are internal status for purpose of h/w debugging.

(4) LIF0 Padding Line Cycle Register (VI6_LIF0_PADLN_CYC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PADLN_CYC[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	PADLN_CYC [5:0]	0	R/W	Cycle of padding pattern. These bits specify the minimum number of cycles of the padding pattern of valid lines and dummy lines. For example, if PADLN_CYC[5:0] is 4, lower 4 bits of PADLN_PATTERN[3:0] indicates padding pattern. A value from 1 to 32 can be specified.

(5) LIF0 Padding Line Pattern Register (VI6_LIF0_PADLN_PT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PADLN_PATTERN[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PADLN_PATTERN[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PADLN_PAT TERN[31:0]	0	R/W	Pattern of padding with dummy lines These bits specify the padding pattern of valid lines and dummy lines. In each bit, 0b indicates an active line, 1b indicates a dummy line. It starts from the lower bit (Bit 0). First line should be valid line (Always set 0 to PADLN_PATTERN[0]).

(6) LIF0 Padding Line Value Register (VI6_LIF0_PADLN_VAL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PADA[7:0]								PADR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PADG[7:0]								PADB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PADA[7:0]	0	R/W	Padding data of EDC code for dummy line.
23 to 16	PADR[7:0]	0	R/W	Padding data of R/Cr for dummy line.
15 to 8	PADG[7:0]	0	R/W	Padding data of G/Y for dummy line.
7 to 0	PADB[7:0]	0	R/W	Padding data of B/Cb for dummy line.

Note: Set H’24FFFFFF to the register, when output format is YUV422. Set H'2EFFFFFF to the register in other case.

(7) LIF0 Padding Line Size Register (VI6_LIF0_PADLN_SIZE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PADLN_VSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	PADLN_VSIZE[12:0]	0	R/W	Vertical Size of LIF output in case of enabling Padding line When VI6_LIF0_CTRL.PADL is 1, dummy lines are included in the output data from LIF, so set the LIF output VSIZE including dummy lines to these bits. If the number of output lines does not reach the value of PADLN_VSIZE [12:0] even if all valid lines are output according to padding pattern, dummy lines are inserted until the number of output lines reaches PADLN_VSIZE [12:0]. ● Output vsize of LIF is same with input vsize to LIF when VI6_LIF0_CTRL.PADL is 0 ● Output vsize of LIF is PADLN_VSIZE [12:0] when VI6_LIF0_CTRL.PADL is 1

33.3.3 DU Registers

33.3.3.1 DU Module Control Register 0 (DU_MCR0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PB_CLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DI_EN	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	PB_CLR	0	R/W	Clear PBUF pointers. This parameter must be changed while VSPD is not running and DU_MSR0.ST_DIF_BSY is “0”. Writing “1”. to this parameter starts clearing the PBUF pointers. After clearing the PBUF pointers is completed, DU_MSR0.ST_PB_WINIT and DU_MSR0.ST_PB_RINIT are read as “1”, this parameter must be written back to “0”.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	DI_EN	0	R/W	Display enable. 1: Enable 0: Disable
7 to 1	—	All 0	R	Reserved
0	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

33.3.3.2 DU Module Status Register 0 (DU_MSR0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	ST_PB_RINIT	ST_PB_RUF	ST_PB_REMPTY	—	ST_PB_WINIT	—	ST_PB_WFULL
Initial Value	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ST_DI_BSY	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	ST_PB_RINIT	1	R	PBUF FIFO read pointer cleared status. 1: Cleared (initial value) 0: Not cleared
21	ST_PB_RUF	0	R	PBUF read underflow status. This parameter clears when PBUF pointers are cleared. 1: Occur PBUF underflow 0: Not occur PBUF underflow
20	ST_PB_REMPTY	1	R	PBUF read empty status. 1: Empty 0: Not empty
19	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	ST_PB_WINIT	1	R	PBUF FIFO write pointer cleared status. 1: Cleared (initial value) 0: Not cleared
17	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	ST_PB_WFULL	0	R	PBUF write full status. 1: Full 0: Not full
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	ST_DI_BSY	0	R	Display I/F status. 1: Busy 0: Idle
7 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

33.3.3.3 DU Module Status Register 1 (DU_MSR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	UF_VACT												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	UF_HACT												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	UF_VACT	H’0000	R	Vactive counter when the PBUF underflow occurs. This parameter is value latched the down counter starting from VACTIVE. This parameter is cleared when DU_MSR0.ST_PB_RUF is cleared.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	UF_HACT	H’0000	R	Hactive counter when the PBUF underflow occurs. This parameter is value latched the down counter starting from HACTIVE. This parameter is cleared when DU_MSR0.ST_PB_RUF is cleared.

33.3.3.4 DU Interrupt Mask Register 0 (DU_IMR0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IM_PB_RUF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	IM_PB_RUF	0	R/W	Mask PBUF read underflow interrupt. 1: Mask interrupt 0: Not mask interrupt

33.3.3.5 DU Display I/F Timing Register 0 (DU_DITR0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	HSPOL	VSPOL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEMD		—	—	—	—	—	—	—	DPI_CLKMD
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	HSPOL	1	R/W	hsync polarity. 1: High active 0: Low active
16	VSPOL	1	R/W	vsync polarity. 1: High active 0: Low active
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	DEMD	11b	R/W	de output mode. 00: Fixed to Low 01: Reserved 10: Reserved 11: Data enable (High active)
7 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	DPI_CLKMD	0	R/W	Display parallel interface clock mode. 0: Through output 1: Inversion output

33.3.3.6 DU Display I/F Timing Register 1 (DU_DITR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VACTIVE												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VSA											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	VACTIVE	H’0000	R/W	The number of lines in the Vactive period. This parameter should be more than 0.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	VSA	H’000	R/W	The number of lines in the Vsync period. This parameter should be 0 or more, and DU_DITR1.VSA + DU_DITR2.VBP should be more than 0. If set 0, Vsync is not asserted to DU_DITR0.VSPOL.

Figure 33.31 shows the relationship between the parameters and the video output timings.

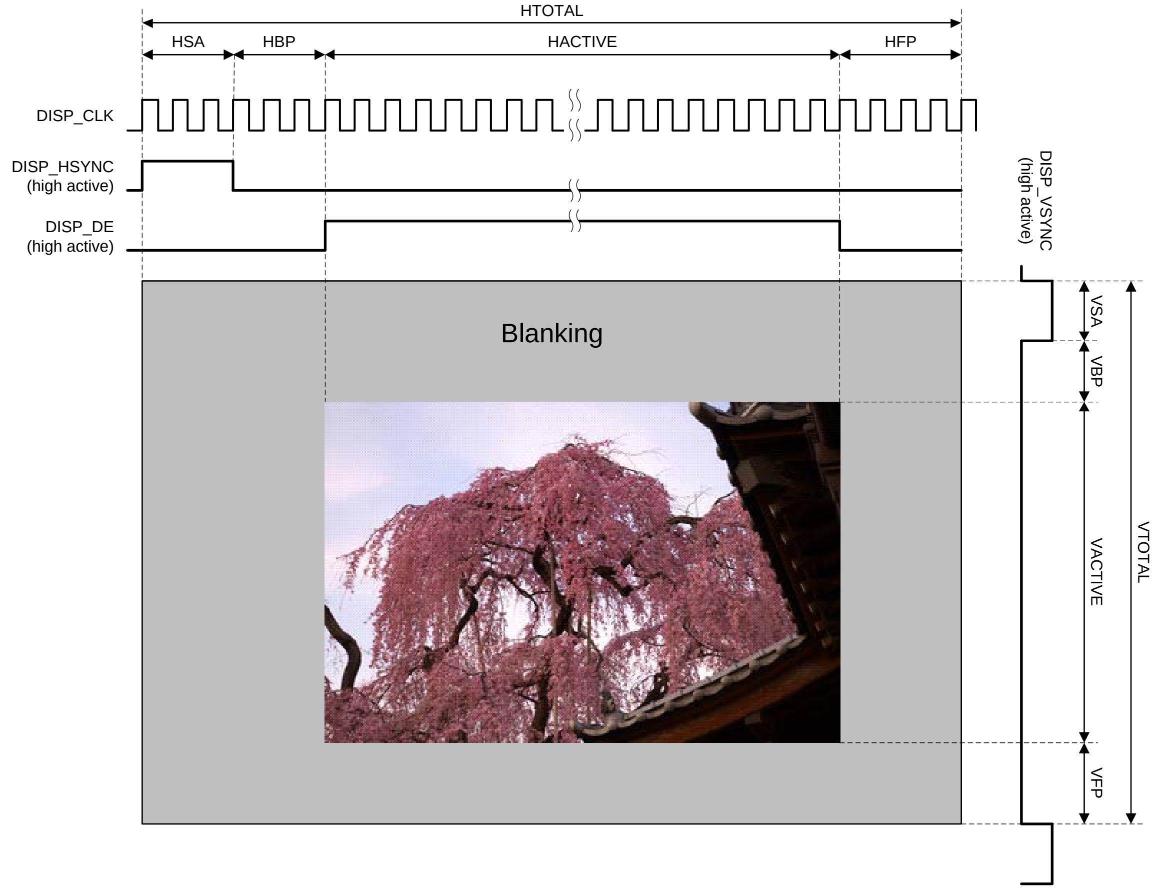


Figure 33.31 Video output timings

33.3.3.7 DU Display I/F Timing Register 2 (DU_DITR2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VFP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VBP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	VFP	H’0000	R/W	The number of lines in the Vfront period. This parameter should be 0 or more.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	VBP	H’0000	R/W	The number of lines in the Vback period. This parameter should be 0 or more, and DU_DITR1.VSA + DU_DITR2.VBP should be more than 0.

33.3.3.8 DU Display I/F Timing Register 3 (DU_DITR3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	HACTIVE												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	HSA											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	HACTIVE	H’0000	R/W	The number of cycles (pixels) in the Hactive period. This parameter should be more than 0.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	HSA	H’000	R/W	The number of cycles in the Hsync period. This parameter should be 0 or more, and DU_DITR3.HSA + DU_DITR4.HBP + DU_DITR4.HFP (=Hblank) should be 3 or more. If set 0, Hsync is not asserted to DU_DITR0.HSPOL.

33.3.3.9 DU Display I/F Timing Register 4 (DU_DITR4)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	HFP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HBP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	HFP	H’0000	R/W	The number of cycles in the Hfront period. This parameter should be 1 or more, and DU_DITR3.HSA + DU_DITR4.HBP + DU_DITR4.HFP (=Hblank) should be 3 or more.
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	HBP	H’0000	R/W	The number of cycles in the Hback period. This parameter should be 0 or more, and DU_DITR3.HSA + DU_DITR4.HBP + DU_DITR4.HFP (=Hblank) should be 3 or more.

33.3.3.10 DU Module Control Register 1 (DU_MCR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PB_AUTOCLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OPMD	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	H'0000	R	Reserved When read, the initial value is read. The written value will be ignored.
16	PB_AUTOCLR	0	R/W	PBUF pointers auto clear enable. Specify timing to clear the PBUF pointers. 0: Not clear until DU_MCR0.PB_CLR is asserted. 1: Clear during blanking (according DU_MCR1.OPMD).
15 to 9	—	H'00	R	Reserved When read, the initial value is read. The written value will be ignored.
8	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 2	—	H'00	R	Reserved When read, the initial value is read. The written value will be ignored.
1, 0	OPMD	All 0	R/W	Operation mode Specify behavior of LIFC and PBUF during Vfront and Hfront period. This parameter must be changed while VSPD is not running. 00: After frame end, LIFC wait, and PBUF pointers are cleared if DU_MCR1.PB_AUTOCLR is set to "1". 01: After line end, LIFC wait, and PBUF pointers are cleared if DU_MCR1.PB_AUTOCLR is set to "1". 10: Reserved 11: LIFC always request to VSPD.PBUF pointers are not cleared until DU_MCR0.PB_CLR is set to "1".

33.3.3.11 DU PBUF Control Register 0 (DU_PBCR0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PB_DEP				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	PB_DEP	H’00	R/W	Valid PBUF depth. This parameter should be set H’1F.

33.3.3.12 DU PBUF Control Register 1 (DU_PBCR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PB_RUFOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	H’0000_0000	R	Reserved When read, the initial value is read. The written value will be ignored.
0	PB_RUFOP	0	R/W	Specify behavior when the PBUF underflow occurs. 0: Continue outputting invalid data 1: Output DU_PBCR2.PB_RUFDAT

33.3.3.13 DU PBUF Control Register 2 (DU_PBCR2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PB_RUFDAT							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB_RUFDAT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved When read, the initial value is read. The written value will be ignored.
23 to 0	PB_RUFDAT	H'00_0000	R/W	Read data when the PBUF underflow occurs.

33.4 Operation

33.4.1 Operation Control Setting

(a) To start image process

FCPVD registers do not need to be set when starting an image process. Therefore, refer the **Section 33.4.4** for the guidance of starting an image process.

33.4.2 Reset Operation

FCPVD reset can be conducted by power on reset. Power on reset initializes the whole FCPVD logic immediately.

33.4.2.1 Power-on Reset, Hardware reset

The power-on reset is controlled by an externally input signal called hardware reset. This reset initializes the whole internal logic of LCDC.

33.4.3 Registers to set fixed value

Set fixed value to following registers in any case.

[1] Always set H’0000 0808 to VI6_CLK_DCSWT

[2] Always set D’256 to VI6_DL_CTRL.AR_WAIT [7:0] in case of using display list.

[3] Always set D’2 to VI6_DL_EXT_CTRLn.POLINT [5:0] in case of using extended display list of WPFn.

[4] Always set D’1 to VI6_DL_EXT_CTRLn.DLPRI in case of using extended display list of WPFn.

[5] Always set D’0 to VI6_DL_EXT_CTRLn.EXPRI in case of using extended display list of WPFn.

[6] Always set H’0000 0500 toVI6_DPR_WPFn_FPORCH (n = 0, 1) in case of using WPFn.

[7] Always set D’1500 to VI6_LIF0_CTRL.OBTH [11:0] in case of using LIF (VI6_LIF0_CTRL.LIF_EN = 1).

[8] Always set H’8600 0000 to VI6_LIF0_LBA in case of using LIF0 (VI6_LIF0_CTRL.LIF_EN = 1).

[9] Always set D’1 to VI6_LIFn_CTRL.REQSEL in case of using LIFn (VI6_LIFn_CTRL.LIF_EN = 1).

[10] Always set D’0 to VI6_LIFn_CSBTH in case of using LIFn (VI6_LIFn_CTRL.LIF_EN = 1).

[11] Always set H'1F to DU_PBCR0 in case of using DU (DU_MCR0.DI_EN = 1).

33.4.4 Concept of VSPD Operation Starting and Stopping

The VSPD provides one channel of image processing. Each channel is started by setting the corresponding start register. Here, starting a processing channel means starting one of WPF0, which are output modules of the VSPD. Use the start registers shown in **Table 33.29** to start WPF modules.

After a WPF module is started and specified processing is completed, the corresponding channel stops operation and notifies the end of processing through an end interrupt. End interrupts are generated through the end interrupt source registers shown in **Table 33.29**; clearing a source register cancels the corresponding interrupt signal. Each of the operating status registers shown in the table indicates the busy state after the corresponding channel is started through the start register until processing is completed and operation stops. **Figure 33.32** shows these operation timings.

Table 33.29 Target Module and Corresponding Registers for Starting and Stopping Operation

Target Module	Start Register	End Interrupt Source Register	Operating Status Register
WPF0	VI6_CMD0.STRCMD	VI6_WPF0_IRQ_STA. FRE	VI6_STATUS.SYS0_ACT

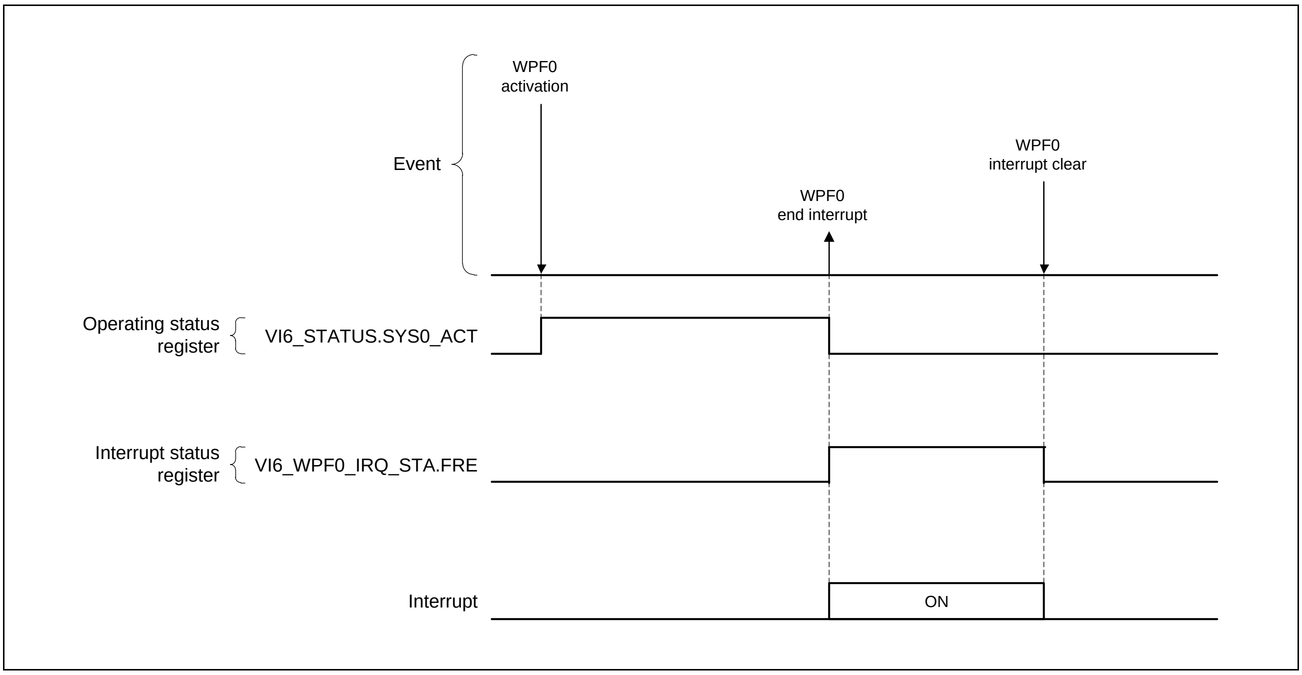


Figure 33.32 VSPD Startup and Status of Each Register and Interrupt

The following describes the operating states (operating or stopped) of VSPD internal modules. As described in **Section 33.3.2**, the VSPD has several image processing modules and the connections between modules are determined by the DPR. Accordingly, the operating state of a module is the same as that of the target WPF for that module. For example, when the target WPF for the BRS is WPF0, the BRS operating state is the same as that of WPF0; that is, the BRS operating state is indicated by the VI6_STATUS.SYS0_ACT as shown in **Table 33.29** and the status change timing is shown as VI6_STATUS. SYS0_ACT in **Figure 33.32**. Likewise, the operating states of all modules connected to WPF0 is indicated by VI6_STATUS. SYS0_ACT.

Connections should be changed through the DPR-related registers (described later) while all modules to be affected by any change in connections are stopped. If connections through the DPR are changed during operation, the VSPD will hang.

Figure 33.33 shows relation between start reservation and operating status. When start is reserved while VSPD is operating, the reservation is accepted after VSPD status is moved from “operating” into “idle” as shown in **Figure 33.33**.

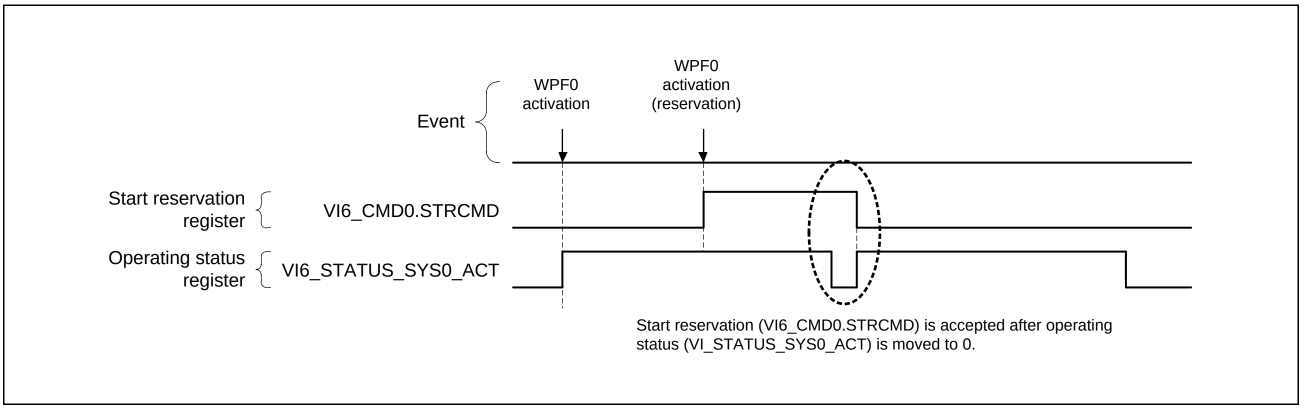


Figure 33.33 VSPD Start reservation and Operating status

33.4.5 Display List

33.4.5.1 Functional Description

The VSPD provides the display list function. As a display list, the VSPD automatically downloads the register settings except for the control registers (**Sections 33.3.2.4** and **Section 33.3.2.5**) from external memory and stores the settings in the VSPD registers. This function is advantageous in that the interrupt processing or register setting modification processing can be executed without CPU intervention during multiple-frame processing because the register settings used for VSPD processing are prepared in advance in external memory such as SDRAM.

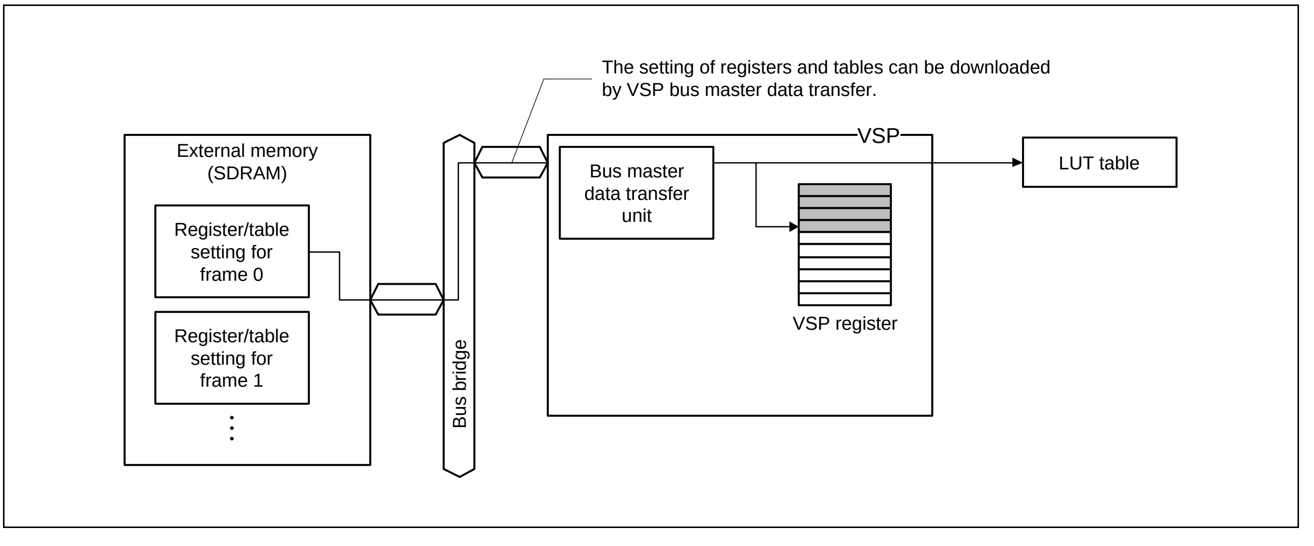


Figure 33.34 Concept of Display List

To use display lists, specify the external memory addresses to the display list control registers described in **Section 33.3.2.5**. The register settings or various types of information should be stored in external memory in the format described in **Section 33.4.5.2**. **Figure 33.35** shows the difference between VSPD operation through normal register settings and through display lists.

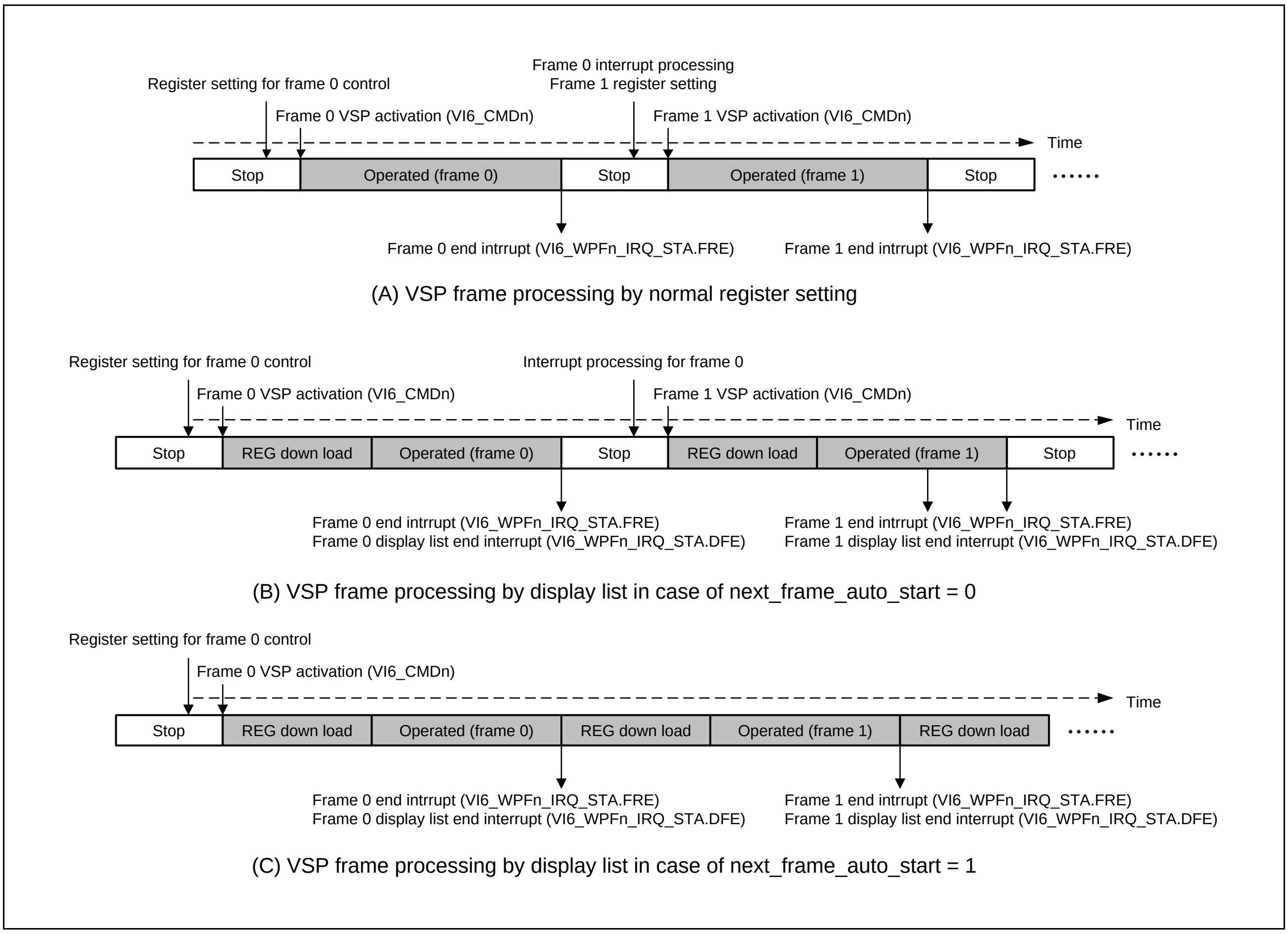


Figure 33.35 Comparison of VSPD Operation between Normal Register Settings and Display Lists

As shown in **Figure 33.35** (A), in the VSPD processing through normal register settings, all registers should be set up before the VSPD is started for each frame. After the VSPD processing is completed, the VSPD outputs a frame end interrupt (VI6_WPFn_IRQ_STA.FRE). This method requires a certain amount of time for register settings or interrupt processing by the CPU between frames. In contrast, when display lists are used, the VSPD automatically downloads register settings from external memory as shown in **Figure 33.35** (B) and (C), which reduces the load on the CPU between frames.

Figure 33.35 (B) shows the display list usage where the VSPD stops at the end of every frame; only the VSPD start processing for each frame is done by the CPU. This is suitable for the cases when the CPU controls synchronization of frame processing in frame buffer management or when the amount of register values or table data to be set in the VSPD is large. In the case shown in **Figure 33.35** (C), as soon as the frame processing ends, the VSPD automatically begins next frame operation and starts downloading new register settings. This is the fastest operation using display lists.

Table 33.30 shows the modes of the display list and the supported functions for each mode. The detail of each mode is described in the following sections.

Table 33.30 Display list mode and supported functions

Mode	Extended Display List	Continuous Frames
Normal Display List Mode	Supported	Controlled by “next frame auto start” in the header of the display list.
Header-less Display List Mode	Not Supported	Controlled by VI6_DL_CTRL.CFM0 bit.

33.4.5.2 Normal Display List Mode

The VSPD display lists include control information as well as simple register settings in order to control multiple-frame processing in an optimum way for each application. **Figure 33.36** shows the display list structure.

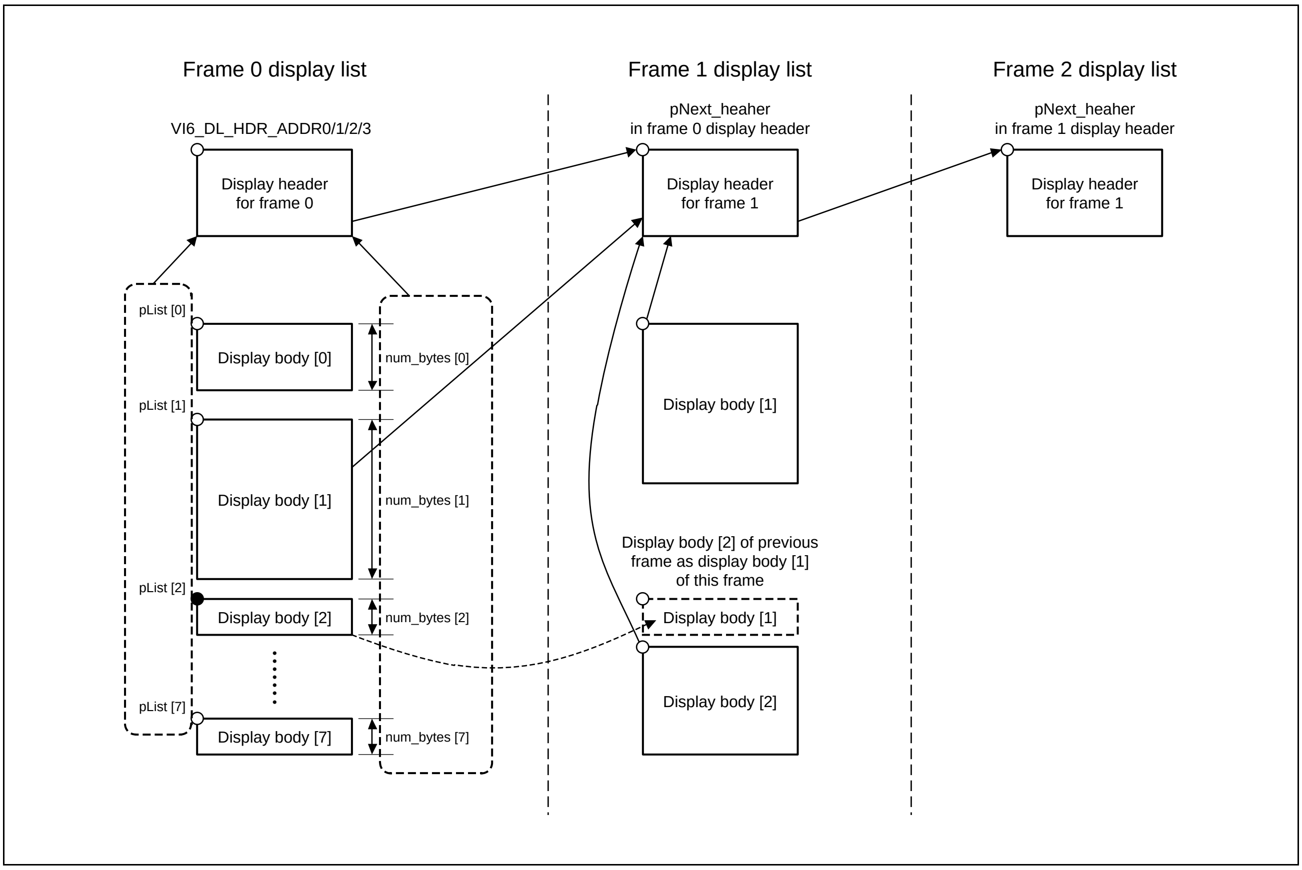


Figure 33.36 Structure and Concept of VSPD Display List

A VSPD display list consists of two sections; a header section for storing various information and control flags and a body section for storing register and table settings. A combination of these two sections is defined as a display list for a frame. The register and table settings can be divided and stored in up to eight separate bodies allocated in memory. Therefore, the separate bodies storing the register settings for one frame (for example, frame 0) can have non-sequential start addresses; that is, the bodies for one frame can be allocated to areas distant from each other in memory. To gather these bodies and configure the register settings for one frame, a header is used. A display header stores the number of bodies linked with the header and the start address and size of each body.

The VSPD analyzes the header to gather register and table settings stored in separate memory areas and reconfigures the complete register settings.

The addresses of display headers should be specified in the VSPD registers described in **Section 33.3.2.5**. When activated in a mode that uses display lists, the VSPD downloads display headers from the addresses specified in VI6_DL_HDR_ADDR0 (number 0 correspond to the WPF channel index numbers), analyzes the numbers of bodies and the address and data size of each body, downloads the bodies, and completes register and table settings. After display list downloading is completed, the VSPD becomes ready for frame image processing; the VSPD then starts the actual frame processing.

After processing of a frame ends, the VSPD proceeds to the next frame processing. Here, there are two modes for starting the next frame processing as shown in **Figure 33.35**. In one mode, the VSPD stops operation and waits for the next activation by the CPU in the same way as when display lists are not used. In this mode, the address used for downloading the display header of the next frame is kept by the internal hardware. Therefore, if valid address information is not stored in the display header for the previous frame, a correct value should be specified in VI6_DL_HDR_ADDR0 while the VSPD is stopped. When the VSPD is started after a correct value is specified, the VSPD starts next frame processing with the same procedure for the previous frame. In contrast to this mode, which stops the VSPD after the end of one-frame processing, there is another mode for automatically starting next frame processing. In automatic start mode, the VSPD downloads the next display header as soon as the previous frame processing ends. After downloading ends, the VSPD starts image processing. The information regarding mode selection, that is, whether to automatically start next frame processing, should be stored in the display header downloaded for the previous frame.

In the automatic start mode, the VSPD continues processing until the display header for a frame specifies that the next frame should not be started automatically. To stop processing during automatic execution, use a software reset (VI6_SRESET).

To strictly define the display list format described above, the following shows the grammatical structure of a display list using pseudo-code. First, to simplify the description of the display list format in the following pages, **Table 33.31** defines a function. Function zero bits (num bits) generates a string of one-bit 0s for the number of bits specified by the parameter for the function. By using this function, **Table 33.32** defines the header section format of a display list and **Table 33.33** defines the body section format and **Table 33.34** defines the extended display list body section format.

Data order of Display List Header Section, Display List Body Section and Extended Display List Body Section are assumed as the data is stored in SDRAM by big endian. **Table 33.35** shows an example of data order. If data order is not same as the definition, data order within 8byte unit can be adjusted by setting VI6_DL_SWAP (see the **Section 33.3.2.5(3)**).

Table 33.31 Definition of a Function for Simple Description

Syntax	Bit Count
zero_bits (num_bits)	
{	
for (i = 0; i<num_bits; i++) {	
zero_bit	1
}	
}	

Bit String	Contents
zero_bit	zero_bit indicates a 1-bit integer having a value of 0.

Table 33.32 Format of Display List Header Section

Syntax	Bit Count
display_header () /* Fixed length */	
{	
zero_bits (29)	
num_list_minus1	3
for (i=0; i<8; i++)	
zero_bits (15)	
num_bytes [i]	17
pList [i]	32
}	
pNext_header	32
zero_bits (23)	
wait_wup	1
zero_bits (3)	
ignore_upd_dl	1
zero_bits (2)	
current_frame_int_enable	1
next_frame_auto_start /* 76 bytes from the beginning of this header*/	1
if (VI6_DL_EXT_CTRL.EXT) {	
zero_bits (32) /* padding zero 4 bytes for alignment */	
zero_bits (6)	
pre_ext_dl_exec	1
post_ext_dl_exec	1
zero_bits (8)	
pre_ext_dl_num_cmd	16
pre_ext_dl_pList	32
zero_bits (16)	
post_ext_dl_num_cmd	16
post_ext_dl_pList /* 96 bytes from the beginning of this header*/	32
}	
}	

Bit String	Contents
num_list_minus1	Specifies the value obtained by subtracting 1 from the total number of display list bodies linked with the display header. For example, when this bit field is set to 0, this display list uses one body.
num_bytes [i]	Specifies the number of bytes in the i-th display list body (indicated by index i). Be sure to specify a multiple of eight bytes. For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0.
pList [i]	Specifies the start address of the i-th display list body (indicated by index i). Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0). For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0.
pNext_header	Specifies the address of the display list header for the next frame. After display list downloading ends, the VSPD keeps its value in the internal memory and uses it in the next display list header downloading. Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0).
wait_wup	When this bit is 1, VSPD starts reading image data from external memory after WUP (Wake Up) signal from any channel of VIN asserts. This bit is available for WPF0.
ignore_upd_dl	If this bit is set to 1, new display list pointed by VI6_DL_HDR_ADDR0 is not downloaded in VSPD H/W for the next frame, even if S/W sets VI6_DL_HDR_ADDR0 during the frame. New display list is downloaded in VSPD H/W at next frame of the processing which this bit is 0. This bit is available for WPF0.
current_frame_int_enable	This is a flag that indicates whether to set the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) to 1 when the current frame processing ends. If this flag is set to 0, the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) is not set to 1 when one-frame processing by this display header ends. In this state, even if the display list end interrupt is enabled (VI6_WPFn_IRQ_ENB.DFEE is set to 1), no interrupt will be generated. If this flag is set to 1, the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) is set to 1 when one-frame processing by this display header ends. In this state, if the display list end interrupt is enabled (VI6_WPFn_IRQ_ENB.DFEE is set to 1), the VSPD generates an interrupt.
Next_frame_auto_start	Enables or disables automatic start of next frame processing when one-frame processing by this display header ends. If this bit is set to 1, the VSPD starts next frame processing as soon as one-frame processing by this display header ends, and starts downloading the next frame display header from the pNext_header address specified in this display header. If this bit is set to 0, the VSPD stops operation when one-frame processing by this display header ends. In this case, start the VSPD through VI6_CMDn to process the next frame.
pre_ext_dl_exec	Enables execution of the extended display list for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. If this bit is set to 1, the VSPD executes the extended display list for frame preprocessing. The VSPD does not execute it if this bit is set to 0. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
post_ext_dl_exec	Set 0 to this bit.
pre_ext_dl_num_cmd	Specifies the number of commands in the extended display list body section for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. The number of commands that can be specified is 1, and a command is 16 bytes. When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
pre_ext_dl_pList	Specifies the start address of the area where the extended display list body section for frame preprocessing is stored when VI6_DL_EXT_CTRL.EXT is 1. Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
post_ext_dl_num_cmd	Set 0 to this bit.
post_ext_dl_pList	Set 0 to this bit.

Table 33.33 Format of Display List Body Section

Syntax	Bit Count
display_list (num_bytes) /* Variable length (num_bytes) */	
{	
for (i =0; i<num_bytes; i+=8) {	
set_address	32
set_data	32
}	
}	

Bit String	Contents
set_address	Specifies the address where the value specified by set_data is to be stored. Specify a register address. Note : Register space of VSPD is 32Kbyte (H'0000 - H'7FFC). So, set 0 to upper 17bit of set_address in Display List Body Section.
set_data	Specifies the value to be stored in the address specified by set_address. Specify a value to be set in a register.

Table 33.34 Format of Extended Display List Body Section

Syntax	Bit Count
ext_dl_display_list (num_llw) /* Variable length (pre/post_ext_dl_num_bytes) */	
{	
for (i=0; i<num_llw; i+=2) {	
ext_dl_cmd	64
ext_dl_data	64
}	
}	

Table 33.35 Data order of display list body

Address	Syntax
pList	set_address [0] (bit31-24)
pList + 1	set_address [0] (bit23-16)
pList + 2	set_address [0] (bit15-8)
pList + 3	set_address [0] (bit7-0)
pList + 4	set_data [0] (bit31-24)
pList + 5	set_data [0] (bit23-16)
pList + 6	set_data [0] (bit15-8)
pList + 7	set_data [0] (bit7-0)
pList + 8	set_address [1] (bit31-24)
pList + 9	set_address [1] (bit23-16)
pList + 10	set_address [1] (bit15-8)
pList + 11	set_address [1] (bit7-0)
pList + 12	set_data [1] (bit31-24)
pList + 13	set_data [1] (bit23-16)
pList + 14	set_data [1] (bit15-8)
pList + 15	set_data [1] (bit7-0)
~~	~~~

33.4.5.3 Header-less Display List Mode

The header-less display list does not have the display header listed in **Table 33.32**, and it has the simplest structure that has only single body. The extended display list function is not available in case of the header-less display list mode. Set the value 1 to VI6_DL_CTRL.NH0 (see the **Section 33.3.2.5(1)**) to use the header-less display list. The start address downloaded by the header-less display list should be set to VI6_DL_HDR_ADDR0. And the size of the display body which is originally defined in the display header should be set to VI6_DL_BODY_SIZE0. The header-less display list is available only in WPF0.

33.4.5.4 Restrictions on Display List Usage

Access to the general control registers and display list control registers through a display list is prohibited. When using display lists, be sure to observe the following restrictions on register access by the CPU.

1. Do not execute write access to the same register (same address) from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSPD is not guaranteed.
2. Do not execute write access to the same LUT lookup table from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSPD is not guaranteed. Here, the same lookup table means the address space having the same space name shown in **Table 33.36**.
3. When read access by the CPU and write access through a display list to the same register (same address) occur at the same time, the read value returned to the CPU is not guaranteed.
4. When read access by the CPU and write access through a display list to the same LUT lookup table occur at the same time, the read value returned to the CPU is not guaranteed. Here, the same lookup table means the address space having the same space name shown in **Table 33.36**.
5. For other restrictions on values and timing of register setting through display lists, refer to the restrictions on normal register settings described in **Section 33.3.2.3**.

6. Manipulation and setting of the registers described in **Section 33.3.2.4** and **Section 33.3.2.5** through a display list is prohibited.
7. The extended display list function enabled by VI6_DL_EXT_CTRL.EXT is to control the registers of the ICB connected with the VSPD. Do not use extended display lists for any other purpose.

33.4.6 Interrupt Processing

When a WPF module generates an internal source that should be notified, an interrupt signal is output. As internal sources are generated in WPF independently, the VSPD has the following registers to control interrupts in WPF.

WPFn Interrupt Enable Registers (VI6_WPFn_IRQ_ENB: n = 0) (refer to **Section 33.3.2.4(12)**)

WPFn Interrupt Status Registers (VI6_WPFn_IRQ_STA: n = 0) (refer to **Section 33.3.2.4(13)**)

CAUTION

To use interrupts, enable them through WPF interrupt enable registers. If an interrupt source register has already been set to 1 for some reason, an unintended interrupt will occur as soon as the corresponding interrupt enable register is set as enabled. To avoid this, before enabling interrupts through WPF interrupt enable registers, be sure to clear all WPF interrupt sources to be enabled to 0. Be careful about this procedure when setting up registers before starting the VSPD.

33.4.7 Lookup Table Settings

33.4.7.1 LUT

For a single entry to the LUT space (see **Table 33.36**) of the VSPD, the LUT data is set by a write access in the format shown in **Figure 33.37**.

The entry address of each space is (start address of the space) + (entry number counting from the base point 0 × 4). For example, the address of entry 7 to the LUT space is (offset address) + H’7000 + 7 × 4 = (offset address) + H’701C.

Table 33.36 shows the spaces for which entries can be made. Note that if the module that references that space is operating, write accesses to the relevant space are prohibited. For example, while RPF2 is operating, accesses to the entire space of VI6_CLUT2_TBL are prohibited. When a read access is made to the relevant space during operation of the referencing module, undefined values will be read out.

The operating/stopped state of each module in **Table 33.36** is the operating state of the WPF to which each module is connected. Determine whether the module is operating or stopped using each WPF operating status bit in the VI6_STATUS register.

Table 33.36 LUT Space Addresses

Space Name	Space Addresses	Entry Count	Module that References the Space in the Left Column
VI6_LUT_TBL	(offset address) + H’7000 to (offset address) + H’73FC	256	LUT

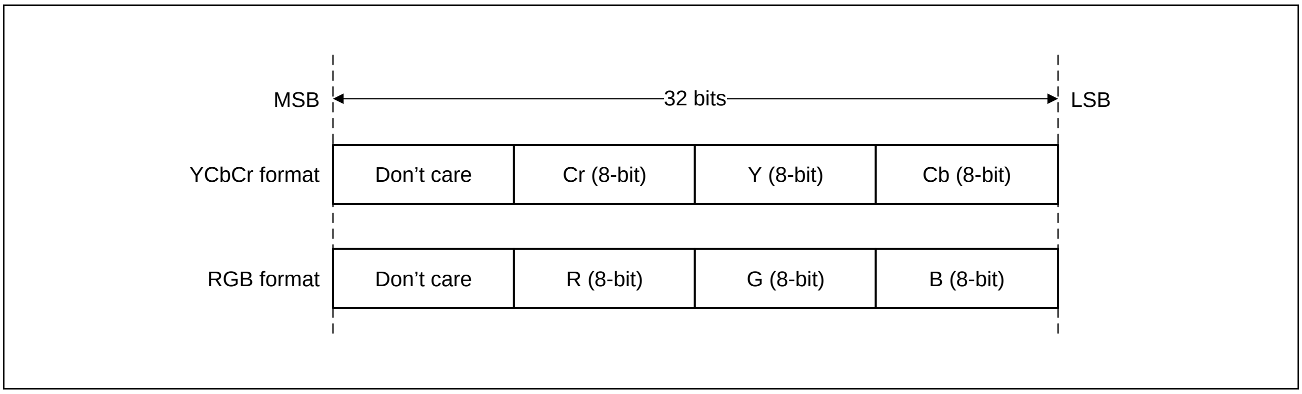


Figure 33.37 VI6_LUT_TBL Formats

33.4.8 Linked with DU

33.4.8.1 Operation flow of VSPD and DU

This section shows operation flow of VSPD by using Normal Display List Mode. For details of Normal Display List Mode, refer to **Section 33.4.5.2**.

Figure 33.38 and **Figure 33.39** shows a procedure (register setting flow) to display image data to display panel.

Figure 33.38 shows setting flow to start VSPD/LIF0 linked with DU. **Figure 33.39** shows setting flow to update display image.

Using display list with enabling next_frame_auto_start shown in **Section 33.4.5** is required to take the flow from **Figure 33.38** and **Figure 33.39**.

NOTE

If display sync operation ([1-8] step in **Figure 33.38**) is set before activating the VSPD, invalid data is displayed until image data read from external memory is ready to be transferred to DU. In this case, image data from external memory is displayed late by one vsync or more.

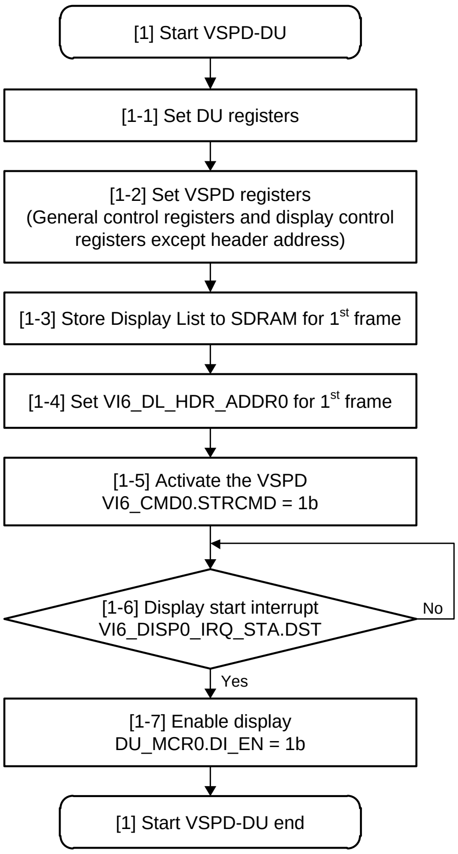


Figure 33.38 Setting flow to start VSPD linked with DU

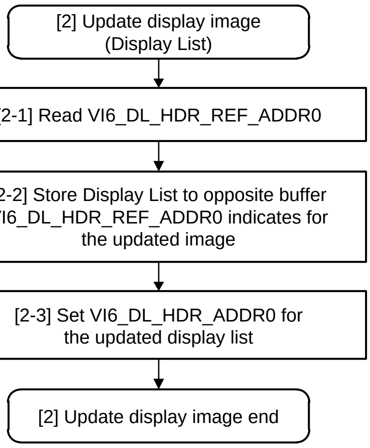


Figure 33.39 Setting flow to update display image

It is possible to update display list at any time (arbitrary timing) as shown in **Figure 33.39**. Don’t overwrite display list in external memory pointed by VI6_DL_HDR_REF_ADDR. It is because VSPD is reading the display list pointed by VI6_DL_HDR_REF_ADDR or will be reading the display list soon.

Detail explanation is shown below.

As shown in **Section 33.3.2.5(6)**, when VSPD is reading display list from external memory (Period [A] in **Figure 33.40**), VI6_DL_HDR_REF_ADDRn indicates header address of the display list referred by VSPD. When VSPD is not reading display list from external memory (Period [B] in **Figure 33.40**), VI6_DL_HDR_REF_ADDRn indicates the value of VI6_DL_HDR_ADDRn, and there is possibility that the display list is being read soon by VSPD at next frame start timing. Therefore, keep display list in external memory pointed by VI6_DL_HDR_REF_ADDR. Detail software sequence to update display list and VSPD -H/W behavior are shown below.

[1] Store latest display list in external memory area different from the area pointed by VI6_DL_HDR_REF_ADDR, because the area pointed by VI6_DL_HDR_REF_ADDR is being read or is being read soon by VSPD.

[2] Set header address of new display list into VI6_DL_HDR_ADDRn.

[3] VSPD reads display list from the external memory area pointed by VI6_DL_HDR_ADDRn at frame start timing.

Refer also to **Section 33.3.2.5(2)** and **Section 33.3.2.5(6)**.

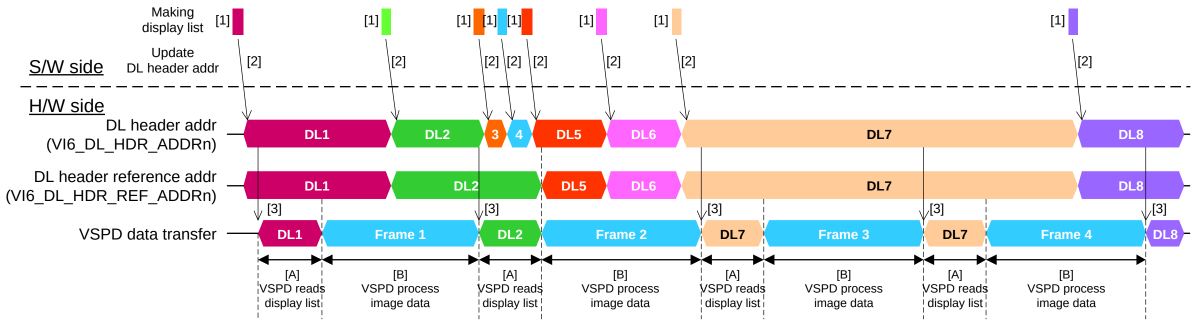
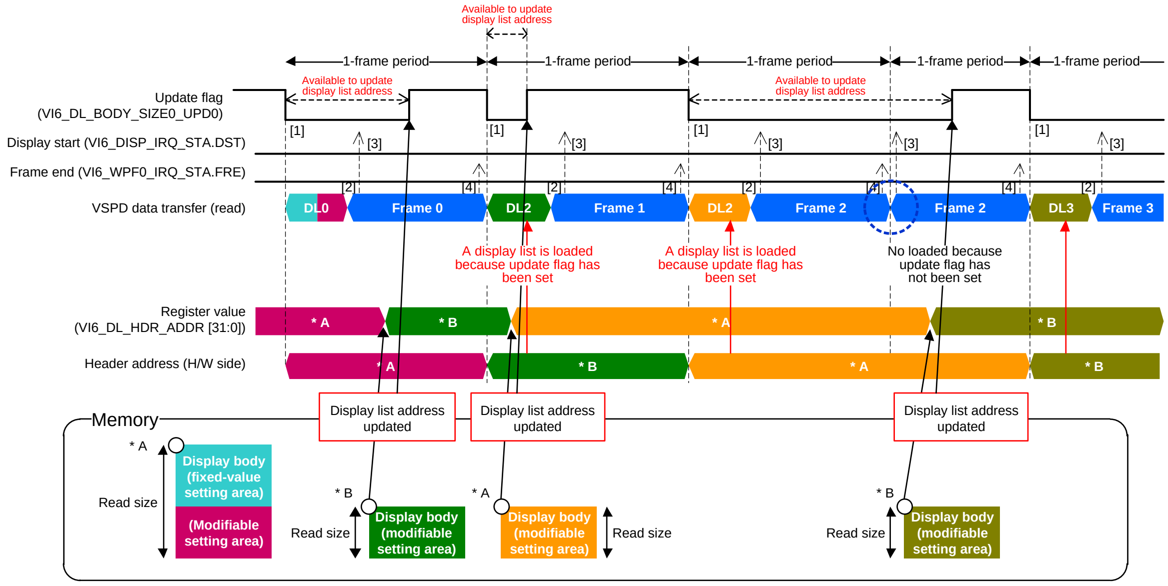


Figure 33.40 Updating display list at arbitrary timing

33.4.8.2 Controlling Two Register Planes Using Display Lists

This section shows operation flow of VSPD by using Header-less Display List Mode. Start procedure and stop procedure are same with Normal Display List Mode in **Section 33.4.8.1** except for the point that VI6_DL_BODY_SIZE0 should be set before step [1-5] in start procedure shown in **Figure 33.38**. The procedure to update display list is different with Normal Display List Mode, and is described later. **Figure 33.41** shows the control of two register planes using header-less display lists (see **Section 33.4.5.3**) and its timing. In the description hereafter, the use of header-less display lists is always assumed and they are simply called display lists.



Display list plane A: “Fixed-value settings” and “modifiable settings” are stored as a set in contiguous area A.
Display list plane B: Only the “modifiable settings” are stored in area B, which is separate from display list A.

To switch settings for every frame, modify two display list planes alternately. After every modification, update the display list address register (VI6_DL_HDR_ADDR0 [31:0]) and writing 1 to update flag (VI6_DL_BODY_SIZE0.UPD0) in the VSPD.
Update flag (VI6_DL_BODY_SIZE0.UPD0) is negated automatically after header address is loaded into H/W side.

Figure 33.41 Controlling Two Register Planes Using a Display List

The VSPD downloads a display list immediately after activation. In the start frame, download the display list that contains all necessary settings. From the next frame on, only the necessary register or table values should be specified in a display list.

When the update flag of VI6_DL_BODY_SIZE0.UPD0 is set to 1, VSPD downloads a new display list at the start of the next frame. When the update flag of VI6_DL_BODY_SIZE0.UPD0 is set to 0, the register settings acquired from the display list previously downloaded are retained and used for the next operation without downloading a new display list.

Order of each VSPD -H/W Events mentioned as [1], [2], [3] and [4] in **Figure 33.41** are shown below.

[1] VI6_DL_BODY_SIZE0.BS0 and VI6_DL_HDR_ADDR0 are downloaded in H/W side. And Update Flag (VI6_DL_BODY_SIZE0.UPD0) is negated automatically.

[2] Reading DisplayList from external memory into VSPD -H/W is finished through master access.

[3] Display start interrupt status (VI6_DISP_IRQ_STA.DST) asserts.

[4] Frame end interrupt status (VI6_WPF0_IRQ_STA.FRE) asserts.

Don’t overwrite the display list in external memory for a period from “[4] VI6_WPF0_IRQ_STA.FRE” to next “[3] VI6_DISP_IRQ_STA.DST”, because VSPD is reading the display list for the period.

33.5 Usage Note

33.5.1 Assignment in Memory Space

Make sure that VSPD memory space shall be mapped to Non-Cache region.

33.5.2 Limitations on Software Reset and Module Standby

Stop LCDC with the software reset sequence following to the guidance in the **Section 33.4.1**.

33.5.3 Input Image Size

Table 33.37 is a list of input size specifications.

Table 33.37 List of Input Size Specifications

Module	Min. Input Size	Max. Input Size	Restriction on Setting Unit
RPF	1 (horizontal) x 1 (vertical) pixel	1920 (horizontal) x 1080 (vertical) pixels	YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. YCbCr420: 2-pixel units both horizontally and vertically. Other formats: 1-pixel units both horizontally and vertically. *Note 1.* When the 1-bpp alpha plane*1 is input, the size can always be specified in 8-pixel units both horizontally and vertically regardless of the input format. *Note 2.* These restrictions including note 1 are applied to the following. VI6_RPFn_SRC_BSIZE VI6_RPFn_SRC_ESIZE
LUT	1 (horizontal) x 1 (vertical) pixel	1920 (horizontal) x 1080 (vertical) pixels	1-pixel units both horizontally and vertically.
BRS	1 (horizontal) x 1 (vertical) pixel	1920 (horizontal) x 1080 (vertical) pixels	1-pixel units both horizontally and vertically.
LIF	1 (horizontal) x 1 (vertical) pixel	1920 (horizontal) x 1080 (vertical) pixels	1-pixel units both horizontally and vertically.
WPF	1 (horizontal) x 1 (vertical) pixel	1920 (horizontal) x 1080 (vertical) pixels	YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. YCbCr420: 2-pixel units both horizontally and vertically. Other formats: 1-pixel units both horizontally and vertically. *Note 1.* This restriction on the WPF only applies to the WPF output size. The WPF input size should be specified in 1-pixel units.

Note 1. When VI6_RPFn_ALPH_SEL.ASEL is set to 011b.

The most important restriction shown in **Table 33.37** is the setting unit. Make appropriate register settings so that the size of the image input to each module does comply with setting units and does not exceed the limits shown in **Table 33.37**.

33.5.4 Output Image Size

The size of the output from each WPF is determined by the results of processing in the modules connected with the DPR. As shown in **Figure 33.1**, the data input to the VSPD is sent to the WPF output modules through the RPF and the modules connected with the DPR. When there is no processing that changes the image size through this data path, the WPF output size is the same as the RPF input size. **Table 33.38** is a list of the processing that changes image size.

Table 33.38 Image Processing that changes image size

Module	Function*1	Related Register	Size of Output from Module
WPF	Input size clipping	VI6_WPFn_HSZCLIP VI6_WPFn_VSZCLIP	When this function is disabled, the input size and the output size are the same. When this function is enabled, the output is in the following size. Horizontal output size: VI6_WPFn_HSZCLIP.HCL_SIZE x 2 setting Vertical output size: VI6_WPFn_VSZCLIP.VCL_SIZE x 2 setting
LIF	Padding line	VI6_LIFn_CTRL	When padding with dummy lines is disabled, output size is same with Input size. When padding with dummy lines is enabled: Horizontal output size: input size Vertical output size: VI6_LIF0_PADLN_SIZE setting

Note 1. For details of each function, refer to the descriptions of the related registers.

Note 2. Refer to **Section 33.3.2.1** for explanation of register bit field.

The input image size can be changed only with the modules and functions shown in **Table 33.38**. With the other modules and functions, the input size and output size are the same. Accordingly, after module connections with the DPR are determined, the VSPD output size can also be determined through the following steps.

1. The image size (VI6_RPFn_SRC_ESIZE) read from the external memory and sent to the DPR by the RPFn is the initial value.
2. When the module connected with the DPR is not a module (function) shown in **Table 33.38**, the output size from the module is the same as the input size; the image size does not need to be updated.
3. When a module connected with the DPR is a module (function) shown in **Table 33.38**, the output image size should be updated to the size shown in the table, which should be used as the input image size for the module connected behind it.
4. When the final image size at the WPFn is determined, this size is the VSPD output size for that WPFn path.

Figure 33.42 shows how to determine the image size in a sample DPR connection through the above steps. The related conditions that determine the image size are also shown.

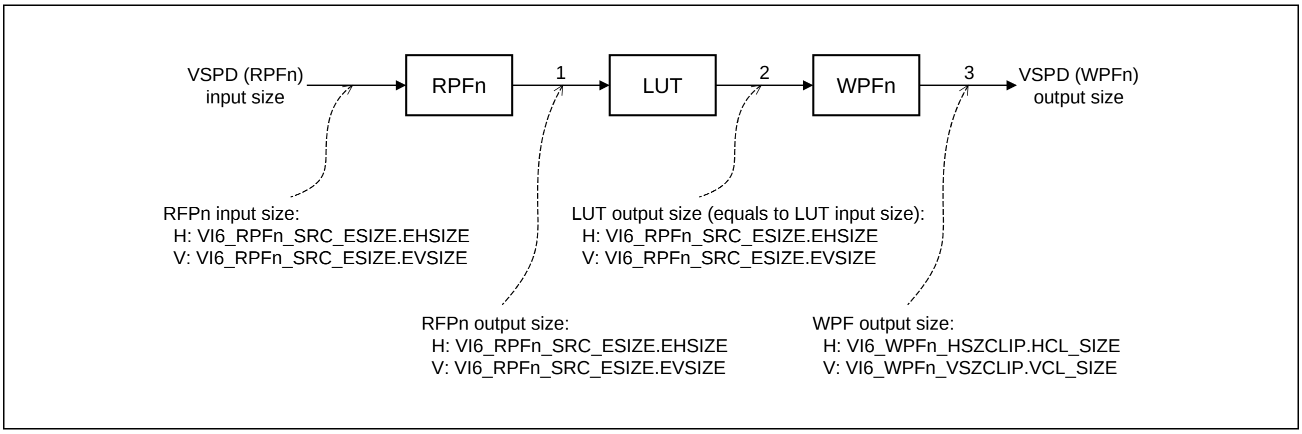


Figure 33.42 Input/output Size for Each Module in a Sample DPR Connection

Make appropriate register settings in each module so that the VSPD output image size determined as shown in the figure does not violate the restrictions shown in **Table 33.38**.

33.5.5 Restriction

When the data output from this unit is written back to the same memory area where the input data for this unit has been read, this unit has the following restrictions.

1. The access order and format on the frame memory are the same between the input pixels and output pixels.
2. Specifying a larger output image size than the input image size, either vertically, horizontally, or both vertically and horizontally, is prohibited.
3. When the YCbCr4:2:0 format is input, operation between color components is prohibited.

These restrictions are summarized in. Refer to the descriptions of the registers related to each restriction. In the table, RPFm indicates the RPFn that inputs the master layer, and WPFwb indicates the WPFn that writes back the output image to the source image area for the master layer.

Table 33.39 Restrictions on Use when Output Data is Written Back to Input Data Area

No.	Restriction	Related Registers
Restriction 1	The RPFm input format and the WPFwb output format should be the same.	VI6_RPFn_INFMT.RDFMT VI6_RPFn_INFMT.VIR
	The RPFm source image storing address and the WPFwb destination address should be the same.	VI6_RPFn_SRCM_ADDR_*
	The RPFm source picture memory stride and the WPFwb destination memory stride should be the same.	VI6_RPFn_SRCM_PSTRIDE VI6_RPFn_SRCM_ASTRIDE
	The RPFm and WPFwb data swapping settings should be the same.	VI6_WPFn_DSWAP
Restriction 2	The RPFm basic read size and extended read size should be the same.	VI6_RPFn_SRC_BSIZE VI6_RPFn_SRC_ESIZE
	Color space conversion is prohibited in RPFm and WPFwb.	VI6_RPFn_INFMT.CSC VI6_WPFn_OUTFMT.CSC
Restriction 3*[1]	NOP should be specified for IROP operation.	VI6_RPFn_ALPH_SEL.IROP
	Color keying is prohibited.	VI6_RPFn_CKEY_CTRL.CV

Note 1. Note: When the input format is not YCbCr4:2:0, restriction 3 in not applied.

34. MIPI DSI

34.1 Overview

This chapter describes the features of the MIPI-DSI unit of this LSI.

This unit is the MIPI-DSI Tx module, it is composed of MIPI DSI-2 Host Controller (LINK), and MIPI D-PHY Tx (D-PHY).

This unit supports MIPI Alliance Specification for Display Serial Interface (DSI) Specification. This unit provides a solution for transmitting MIPI DSI compliant digital video and packets. Normative References are below.

- MIPI Alliance Specification for Display Serial Interface Version 1.3.1
- MIPI Alliance Specification for D-PHY Version 2.1

34.1.1 Features

The following is key features of this unit.

In this document the word “DSI” means “Display Serial Interface Version 1.3.1”.

■ Overview

- 1 channel
- The number of Lane: 4-lane
- Support up to Full HD (1920 × 1080), 60 fps (RGB888)
- Maximum Bandwidth: 1.5 Gbps per lane
- Support Output Data Format: RGB666 / RGB888

■ Protocol Interface

- PPI Interface (8bit)
- Packet generation from Video Input signal for Video Mode (Video-Input Operation)
  - Supports RGB format (18bit, 18bit Loosely and 24bit)
  - Not supports RGB format (30bit and 36bit)
  - Not supports YCbCr 4:2:2 format (16bit, 20bit Loosely and 24bit)
  - Not supports YCbCr 4:2:0 format (12bit)
  - Not supports Compressed Pixel Stream
  - Supports Blanking Packet or LP-11 selection during each of HSA, HBP and HFP blanking interval
  - Supports “Non-Burst Mode with Sync Pulse”, “Non-Burst Mode with Sync Event” and “Burst Mode”
  - Not supports interlaced video
  - Not supports deskew pattern insertion after vertical video data timing
- LP only packet generation and LP packet reception from descriptor list (Sequence Operation channel 0)
  - Supports one sequence input channel
- HS or LP packet generation and LP packet reception from descriptor list (Sequence Operation channel 1)
  - Supports one sequence input channel

■ Link Layer

- Supports 1, 2, 3 and 4 lane configurations
- Supports unidirectional high-speed mode Tx
- Supports bidirectional LP mode Tx/Rx (Only Lane0)
- Supports ECC/Checksum generation for Tx packet
- Supports ECC/Checksum verification and ECC error correction for Rx packet
- Supports automated insertion of EoTp in HS mode
- Supports ULPS (Tx)
- Supports automated power change to LP mode and return to HS mode
- Supports automated clock stop and resume (non-continuous clock mode)
- Supports assignment for Virtual Channel for video input channel in Video-Input Operation
- Supports assignment for individual Virtual Channel for each packet in Sequence Operation
- Supports detection for PHY contention error and timeout error
- Supports generation of scrambled packets
- Not supports input of TE signal

■ PHY Layer

- Number of Lanes
  - Data Lane (DL): 4 Lanes
  - Clock Lane (CL): 1 Lane
- Transfer Rate
  - High Speed (HS) Mode: 80 – 1500Mbps
- Lane Module Functions
  - DL: CIL-MFAA (HS-TX, LP-TX, LP-RX, LP-CD)
  - CL: CIL-MCNN (HS-TX, LP-TX)
- Optional Functions
  - Escape Mode
    - LPDT
      - LP Transmit: Supported
      - LP Receive: Supported
      - ULPS: Supported
      - Reset Trigger: Supported
    - Bi-Direction
      - Bus Turn-Around: Supported
      - Contention Detection: Supported
    - Calibration
      - Deskew Calibration / Preamble: Not-Supported

    - Pre/De-Emphasis: Not-Supported
    - Equalization: Not-Supported
  - Others
    - HS-IDLE: Not-Supported
    - 16bit, 32bit width of Tx Data: Not-Supported
- Power Supply Voltage (at Core Power Pins, Min/Typ/Max)
  - VDD: Same with the condition of this LSI
  - VCCQLV18: Min:1.65V – Typ:1.80V – Max:1.95V

34.1.2 Block Diagram

Figure 34.1 shows a block diagram.

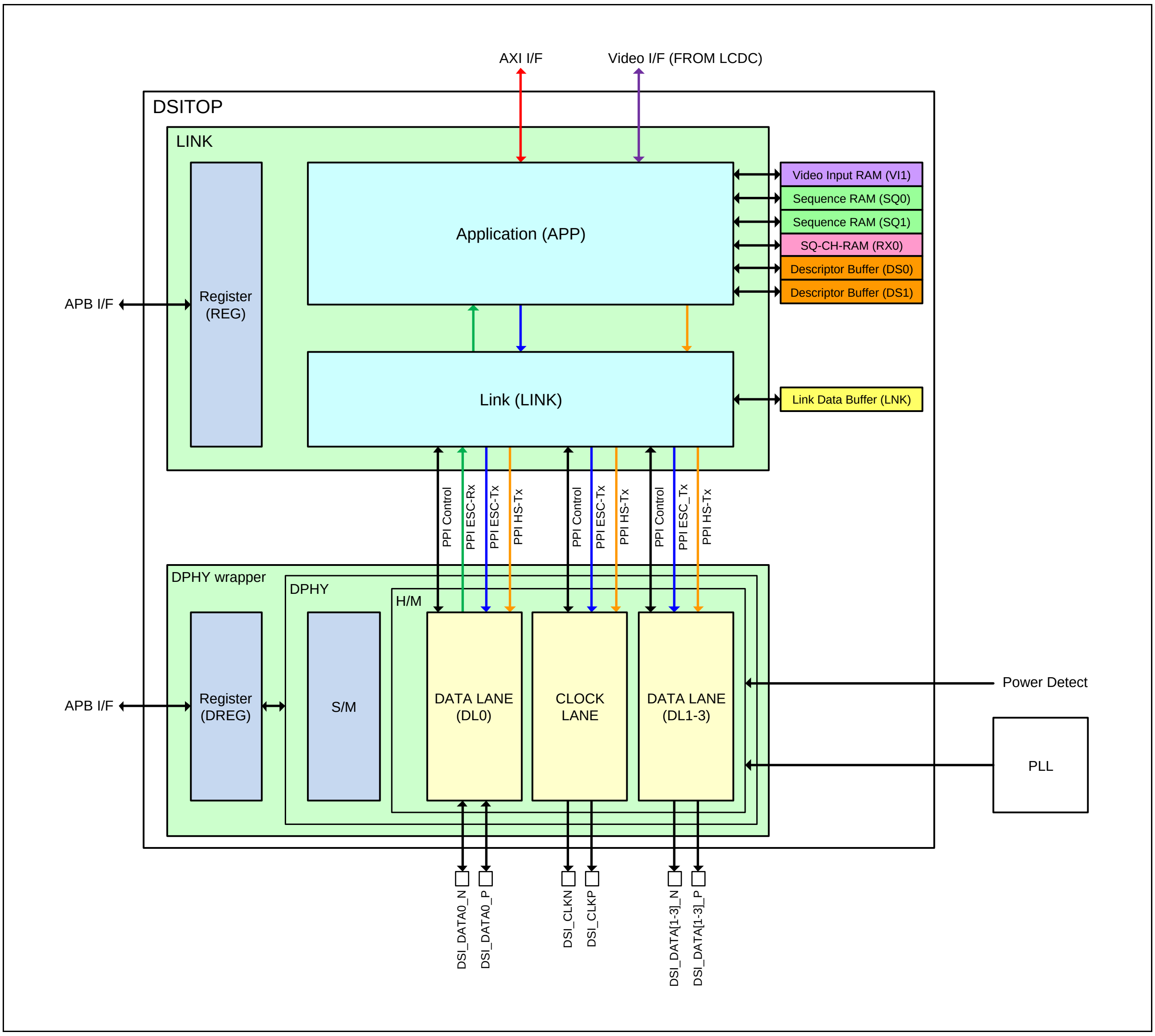


Figure 34.1 Block Diagram

34.1.3 Clock

Table 34.1 shows a list of clocks for this unit.

- T_{LPX}: Transmitted length of any Low-Power state period (Min: 50ns)
- $T_{LPX(H)}$: TLPX of Host Processor (the DSI-2 TX module)
- $T_{LPX(P)}$: TLPX of Peripheral device (connected device)

Table 34.1 Clocks and the frequency range

Clock name	I/O	Description	Frequency (MHz)
MIPI_DSI_PCLK	I	APB clock.	100/50/25/12.5/3.125
MIPI_DSI_ACLK	I	AXI clock.	200/100/50/25/6.25
MIPI_DSI_VCLK	I	Video clock.	148.5~5.803
MIPI_DSI_LPCLK	I	DSI Escape mode Transmit Clock (TxClkEsc). Set D-PHY TxClkEsc (1/$T_{LPX(H)}$) frequency. $T_{LPX(H)}$/$T_{LPX(P)}$ needs to set between 2/3 to 3/2.	16.656/8.328/4.164/2.082
MIPI_DSI_SYSCLK	I	System clock for D-PHY.	133.25
MIPI_DSI_PLLCLK	I	PLL Multiplied clock. {Transfer rate x 4} MHz	160~3000
hsclk	—	DSI High-Speed Transmit Word Clock (TxWordClkHS). Upper limit for D-PHY 1.5Gbps. Lower limit for D-PHY 80Mbps. This clock is supplied from D-PHY directly, and its frequency is 1/16 for MIPI_DSI_PLLCLK.	10 to 187.5
lppclk	—	DSI Escape mode Receive Clock (RxClkEsc). This clock is supplied from D-PHY directly, and its frequency is half of (1/$T_{LPX(P)}$). This clock is generated dividing in D-PHY.	2 to 10

Video clock and DSI HS Byte clock must follow the relationship.

Video clock Frequency [Hz] × Video Pixel Bit Depth[bit]

<= DSI HS Byte clock Frequency [Hz] × 8 [bit] × Number of DSI HS Data Lane

34.1.4 External Pins

Table 34.2 shows the pin configuration.

Table 34.2 External Pins

Pin Name	Input/Output	Function
DSI_CLKP	Output	MIPI DSI clock lane differential clock (pos)
DSI_CLKN	Output	MIPI DSI clock lane differential clock (neg)
DSI_DATA0_P	Output	MIPI DSI data lane0 differential data (pos)
DSI_DATA0_N	Output	MIPI DSI data lane0 differential data (neg)
DSI_DATA1_P	Output	MIPI DSI data lane1 differential data (pos)
DSI_DATA1_N	Output	MIPI DSI data lane1 differential data (neg)
DSI_DATA2_P	Output	MIPI DSI data lane2 differential data (pos)
DSI_DATA2_N	Output	MIPI DSI data lane2 differential data (neg)
DSI_DATA3_P	Output	MIPI DSI data lane3 differential data (pos)
DSI_DATA3_N	Output	MIPI DSI data lane3 differential data (neg)

34.2 Register Configuration

34.2.1 Registers in LINK

This section describes about registers in LINK. All registers can access by 8/16/32 bit.

Base Address: H’0_1086_0000 (Cortex-A55 Address Space)
Base Address: H’4086_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’5086_0000 (Cortex-M33 Address Space Secure)

Table 34.3 Resister list (1/3)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt Status Register	ISR	*2	H’0000_0000	H’000	8/16/32
Link Status Register	LINKSR	*2	H’0000_0000	H’010	8/16/32
Tx Set Register	TXSETR	*2	H’0003_0003	H’100	8/16/32
HS Clock Set Register	HSCLKSETR	*2	H’0000_0000	H’104	8/16/32
ULPS Set Register	ULPSSETR	*2	H’0000_00A0	H’108	8/16/32
ULPS Control Register	ULPSCR	*2	H’0000_0000	H’10C	8/16/32
Reset Control Register	RSTCR	*2	H’0000_0000	H’110	8/16/32
Reset Status Register	RSTSR	*2	H’0000_0000	H’114	8/16/32
DSI Set Register	DSISETR	*2	H’80F1_0001	H’120	8/16/32
Receive Buffer Size Register	RXBUFSZR	*2	H’0000_0000	H’124	8/16/32
Tx Packet Payload Data 0 Register	TXPPD0R	*2	H’0000_0000	H’160	8/16/32
Tx Packet Payload Data 1 Register	TXPPD1R	*2	H’0000_0000	H’164	8/16/32
Tx Packet Payload Data 2 Register	TXPPD2R	*2	H’0000_0000	H’168	8/16/32
Tx Packet Payload Data 3 Register	TXPPD3R	*2	H’0000_0000	H’16C	8/16/32
Rx Status Register	RXSR	*2	H’0000_0000	H’200	8/16/32
Rx Status Clear Register	RXSCR	*2	H’0000_0000	H’204	8/16/32
Rx Interrupt Enable Register	RXIER	*2	H’0000_0000	H’208	8/16/32
Peripheral Response Timeout BTA Set Register	PRESPTOBTASETR	*2	H’0000_0000	H’210	8/16/32
Peripheral Response Timeout LP Set Register	PRESPTOLPSETR	*2	H’0000_0000	H’214	8/16/32
Peripheral Response Timeout HS Set Register	PRESPTOHSSETR	*2	H’0000_0000	H’218	8/16/32
Acknowledge and Error Report Packet Parameter Latest Info Register	AKEPLATIR	*2	H’0000_0000	H’220	8/16/32
Acknowledge and Error Report Packet Parameter Accumulate Status Register	AKEPACMSR	*2	H’0000_0000	H’224	8/16/32
Acknowledge and Error Report Packet Parameter Status Clear Register	AKEPSCR	*2	H’0000_0000	H’228	8/16/32
Rx Result Saved Status Register	RXRSSR	*2	H’0000_0000	H’230	8/16/32
Rx Result Saved Status Clear Register	RXRSSCR	*2	H’0000_0000	H’234	8/16/32
Rx Result Info Overwrite Status Register	RXRINFOOWSR	*2	H’0000_0000	H’238	8/16/32
Rx Result Info Overwrite Status Clear Register	RXRINFOOWSCR	*2	H’0000_0000	H’23C	8/16/32
Rx Result Save Slot 0 Register	RXRSS0R	*2	H’0000_0000	H’240	8/16/32
Rx Result Save Slot 1 Register	RXRSS1R	*2	H’0000_0000	H’244	8/16/32
Rx Result Save Slot 2 Register	RXRSS2R	*2	H’0000_0000	H’248	8/16/32
Rx Result Save Slot 3 Register	RXRSS3R	*2	H’0000_0000	H’24C	8/16/32
Rx Packet Payload Data 0 Register	RXPPD0R	*2	H’0000_0000	H’2C0	8/16/32

Table 34.3 Resister list (2/3)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Rx Packet Payload Data 1 Register	RXPPD1R	*2	H’0000_0000	H’2C4	8/16/32
Rx Packet Payload Data 2 Register	RXPPD2R	*2	H’0000_0000	H’2C8	8/16/32
Rx Packet Payload Data 3 Register	RXPPD3R	*2	H’0000_0000	H’2CC	8/16/32
HSTX Timeout Set Register	HSTXTOSETR	*2	H’0000_0000	H’2E0	8/16/32
LRX-H Timeout Set Register	LRXHTOSETR	*2	H’0000_0000	H’2E4	8/16/32
TA Timeout Set Register	TATOSETR	*2	H’0000_0000	H’2E8	8/16/32
Fatal Error Status Register	FERRSR	*2	H’0000_0000	H’300	8/16/32
Fatal Error Status Clear Register	FERRSCR	*2	H’0000_0000	H’304	8/16/32
Fatal Error Interrupt Enable Register	FERRIER	*2	H’0000_0000	H’308	8/16/32
Clock Lane Stop Time Set Register	CLSTPTSETR	*2	H’0000_0000	H’314	8/16/32
LP Transition Time Set Register	LPTRNSTSETR	*2	H’0000_0000	H’318	8/16/32
Physical Lane Status Register	PLSR	*2	H’0000_00F1	H’320	8/16/32
Physical Lane Status Clear Register	PLSCR	*2	H’0000_0000	H’324	8/16/32
Physical Lane Interrupt Enable Register	PLIER	*2	H’0000_0000	H’328	8/16/32
Video-Input Channel 1 Set 0 Register	VICH1SET0R	*2	H’0000_0000	H’400	8/16/32
Video-Input Channel 1 Set 1 Register	VICH1SET1R	*2	H’1063_0000	H’404	8/16/32
Video-Input Channel 1 Status Register	VICH1SR	*2	H’0000_0000	H’410	8/16/32
Video-Input Channel 1 Status Clear Register	VICH1SCR	*2	H’0000_0000	H’414	8/16/32
Video-Input Channel 1 Interrupt Enable Register	VICH1IER	*2	H’0000_0000	H’418	8/16/32
Video-Input Channel 1 Pixel Packet Set Register	VICH1PPSETR	*2	H’000E_0000	H’420	8/16/32
Video-Input Channel 1 Vertical Size Set Register	VICH1VSSETR	*2	H’0000_0000	H’428	8/16/32
Video-Input Channel 1 Vertical Porch Set Register	VICH1VPSETR	*2	H’0000_0000	H’42C	8/16/32
Video-Input Channel 1 Horizontal Size Set Register	VICH1HSSETR	*2	H’0000_0000	H’430	8/16/32
Video-Input Channel 1 Horizontal Porch Set Register	VICH1HPSETR	*2	H’0000_0000	H’434	8/16/32
Sequence Channel 0 Set 0 Register	SQCH0SET0R	*2	H’0080_0000	H’5C0	8/16/32
Sequence Channel 0 Set 1 Register	SQCH0SET1R	*2	H’0800_0000	H’5C4	8/16/32
Sequence Channel 0 Status Register	SQCH0SR	*2	H’0000_0000	H’5D0	8/16/32
Sequence Channel 0 Status Clear Register	SQCH0SCR	*2	H’0000_0000	H’5D4	8/16/32
Sequence Channel 0 Interrupt Enable Register	SQCH0IER	*2	H’0000_0000	H’5D8	8/16/32
Sequence Channel 1 Set 0 Register	SQCH1SET0R	*2	H’0080_0000	H’600	8/16/32
Sequence Channel 1 Set 1 Register	SQCH1SET1R	*2	H’0830_0000	H’604	8/16/32
Sequence Channel 1 Status Register	SQCH1SR	*2	H’0000_0000	H’610	8/16/32
Sequence Channel 1 Status Clear Register	SQCH1SCR	*2	H’0000_0000	H’614	8/16/32
Sequence Channel 1 Interrupt Enable Register	SQCH1IER	*2	H’0000_0000	H’618	8/16/32
Sequence Channel 0 Descriptor 00-A Register	SQCH0DSC00AR	*2	*1	H’780	32
Sequence Channel 0 Descriptor 00-B Register	SQCH0DSC00BR	*2	*1	H’784	32
Sequence Channel 0 Descriptor 00-C Register	SQCH0DSC00CR	*2	*1	H’788	32

Table 34.3 Resister list (3/3)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Sequence Channel 0 Descriptor 00-D Register	SQCH0DSC00DR	*2	*1	H'78C	32
Sequence Channel 0 Descriptor 01-A Register	SQCH0DSC01AR	*2	*1	H'790	32
Sequence Channel 0 Descriptor 01-B Register	SQCH0DSC01BR	*2	*1	H'794	32
Sequence Channel 0 Descriptor 01-C Register	SQCH0DSC01CR	*2	*1	H'798	32
Sequence Channel 0 Descriptor 01-D Register	SQCH0DSC01DR	*2	*1	H'79C	32
…	…		…		32
Sequence Channel 0 Descriptor 07-A Register	SQCH0DSC07AR	*2	*1	H'7F0	32
Sequence Channel 0 Descriptor 07-B Register	SQCH0DSC07BR	*2	*1	H'7F4	32
Sequence Channel 0 Descriptor 07-C Register	SQCH0DSC07CR	*2	*1	H'7F8	32
Sequence Channel 0 Descriptor 07-D Register	SQCH0DSC07DR	*2	*1	H'7FC	32
Sequence Channel 1 Descriptor 00-A Register	SQCH1DSC00AR	*2	*1	H'800	32
Sequence Channel 1 Descriptor 00-B Register	SQCH1DSC00BR	*2	*1	H'804	32
Sequence Channel 1 Descriptor 00-C Register	SQCH1DSC00CR	*2	*1	H'808	32
Sequence Channel 1 Descriptor 00-D Register	SQCH1DSC00DR	*2	*1	H'80C	32
…	…		…		32
Sequence Channel 1 Descriptor 07-A Register	SQCH1DSC07AR	*2	*1	H'870	32
Sequence Channel 1 Descriptor 07-B Register	SQCH1DSC07BR	*2	*1	H'874	32
Sequence Channel 1 Descriptor 07-C Register	SQCH1DSC07CR	*2	*1	H'878	32
Sequence Channel 1 Descriptor 07-D Register	SQCH1DSC07DR	*2	*1	H'87C	32

Note: Other addresses are Reserved. Do not Access to them.

Note 1. Initial Values are determined by initial value of Descriptor RAM.

Note 2. Refer to the respective register description and **Table 34.5**.

34.2.2 Registers in D-PHY

This section describes about registers in D-PHY. All registers can access by 8/16/32 bit.

Base Address: H’0_1085_0000 (Cortex-A55 Address Space)
Base Address: H’4085_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’5085_0000 (Cortex-M33 Address Space Secure)

Table 34.4 Resister list

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
D-PHY Control Resister 0	DSIDPHYCTRL0	RW	H’00010104	H’00	8/16/32
D-PHY Timing Resister 0	DSIDPHYTIM0	RW	H’01002710	H’04	8/16/32
D-PHY Timing Resister 1	DSIDPHYTIM1	RW	H’04030909	H’08	8/16/32
D-PHY Timing Resister 2	DSIDPHYTIM2	RW	H’05100119	H’0C	8/16/32
D-PHY Timing Resister 3	DSIDPHYTIM3	RW	H’0409060B	H’10	8/16/32

34.3 Register Descriptions

34.3.1 Resisters in LINK

Legend for Register Description.

Table 34.5 Resister R/W

R/W	Write Access	Read Access
R	Ignored	Valid
RW	Valid	Valid
R0W	Valid	The read value is always 0.

34.3.1.1 Interrupt Status Register (ISR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PPI	—	—	—	FERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RCV	—	—	—	VIN1	—	—	—	SQ1	—	—	—	SQ0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	—	0	R	Reserved When read, the initial value is read. The written value will be ignored.
27 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	PPI	0	R	PPI Interrupt When this value is “1”, an interrupt of DSI D-PHY PPI related exists. Refer to the description of PLSR register for more information.
19 to 17	—	All 0	R	Reserved
16	FERR	0	R	FERR Interrupt When this value is “1”, an interrupt of DSI Fatal Error exists. Refer to the description of FERRSR register for more information.
15 to 13	—	All 0	R	Reserved
12	RCV	0	R	RCV Interrupt When this value is “1”, an interrupt of DSI Packet Receive exists. Refer to the description of RXSR register for more information.
11 to 9	—	All 0	R	Reserved
8	VIN1	0	R	VIN1 Interrupt When this value is “1”, an interrupt of Video Input-Operation channel 1 exists. Refer to the description of VICH1SR register for more information.
7 to 5	—	All 0	R	Reserved
4	SQ1	0	R	SQ1 Interrupt When this value is “1”, an interrupt of Sequence Operation channel 1 exists. Refer to the description of SQCH1SR register for more information.
3 to 1	—	All 0	R	Reserved
0	SQ0	0	R	SQ1 Interrupt When this value is “1”, an interrupt of Sequence Operation channel 0 exists. Refer to the description of SQCH0SR register for more information.

34.3.1.2 Link Status Register (LINKSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	LPBUSY	HSBUSY	—	—	—	VICHRUN1	—	—	—	SQCHRUN1	—	—	—	SQCHRUN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
13	LPBUSY	0	R	LP Operation Busy When this value is “1”, it indicates an operation related to LP is running.
12	HSBUSY	0	R	HS Operation Busy When this value is “1”, it indicates an operation related to HS is running.
11 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	VICHRUN1	0	R	Video-Input Channel 1 Running This bit is copy of VICH1SR.RUNNING.
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	SQCHRUN1	0	R	Sequence Channel 1 Running This bit is copy of SQCH1SR.RUNNING.
3 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SQCHRUN0	0	R	Sequence Channel 0 Running This bit is copy of SQCH0SR.RUNNING.

34.3.1.3 Tx Set Register (TXSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NUMLANECAP	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DLEN	CLEN	—	—	—	—	—	—	NUMLANEUSE	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17 to 16	NUMLANECAP	H’3	R	Number of Lanes Capability H’3 (fixed value): Maximum Lane count is 4 (Lane0, Lane1, Lane2 and Lane3 can be used)
15 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	DLEN	0	RW	Data Lane Enable When this bit is “1”, Data Lane is enabled by PHY interface. Be enabled lanes are determined by NUMLANEUSE.
8	CLEN	0	RW	Clock Lane Enable When this bit is “1”, Clock Lane is enabled by PHY interface.
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	NUMLANEUSE	H’3	RW	Number of Lanes for Use Set Lane count for use. Selected Data Lanes are Enabled if DLEN is “1”. H’0: 1Lane (Use of Lane0) H’1: 2Lane (Use of Lane0 and Lane1) H’2: 3Lane (Use of Lane0, Lane1 and Lane2) H’3: 4Lane (Use of Lane0, Lane1, Lane2 and Lane3) Setting value over NUMLANECAP is prohibited.

34.3.1.4 HS Clock Set Register (HSCLKSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	HSCLK MODE	HSCLK RUN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	HSCLKMODE	0	RW	HS Clock running Mode 0b: Request for clock lane HS transmission when HS Transmission scheduled. (non-continuous clock mode) 1b: Keep clock lane to HS transmission. (continuous clock mode) If HSCLKRUN=0, this field has no meaning. Prohibited to change this bit during HSCLKRUN=1.
0	HSCLKRUN	0	RW	Start HS Clock running on Clock Lane 0b: No HS transmission (keep LP) 1b: Allow HS transmission according to HSCLKMODE Confirm stable TxWordClkHS(hsclk) before setting this field to 1. Setting 1 to this field during TXSETR.CLEN=0 is prohibited.

34.3.1.5 ULPS Set Register (ULPSSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ULPSWKUP							
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	ULPSWKUP	H'A0	RW	Twakeup period of ULPS This period is specified by count value of MIPI_DSI_LPCLK *128. This period should be more than 1ms. For example; ● In case MIPI_DSI_LPCLK is 4/3MHz, MIPI_DSI_LPCLK *128=96us, Setting ULPSWKUP = H'0B, Twakeup period = 1.06ms (11*96us) ● In case MIPI_DSI_LPCLK is 20MHz, MIPI_DSI_LPCLK *128=6.4us, Setting ULPSWKUP = H'A0, Twakeup period = 1.02ms (160*6.4us) Setting 1 to this field during TXSETR.CLEN=0 is prohibited.

34.3.1.6 ULPS Control Register (ULPSCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DLULP SEXT	DLULP SENT	—	—	CLULP SEXT	CLULP SENT	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R0W	R0W	R	R	R0W	R0W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	DLULPSEXT	0	R0W	DL ULPS Exit When write “1”, request to all enabled data lane exit from Ultra-low Power State (ULPS). Write “1” when enabled data lanes are not in ULPS, is prohibited. Writing “0” has no effect.
28	DLULPSENT	0	R0W	DL ULPS Enter When write “1”, request to enabled data lane transition to Ultra-low Power State (ULPS). Write “1” when enabled data lanes are in ULPS, is prohibited. Writing “0” has no effect.
27, 26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	CLULPSEXT	0	R0W	CL ULPS Exit When write “1”, request to clock lane exit from Ultra-low Power State (ULPS). Write “1” when clock lane is not in ULPS, is prohibited. Writing “0” has no effect.
24	CLULPSENT	0	R0W	CL ULPS Enter When write “1”, request to clock lane transition to Ultra-low Power State (ULPS). Before setting this field, HSCLKSETR.HSCLKRUN must be set to 0. Write “1” when clock lane is in ULPS, is prohibited. Writing “0” has no effect.
23 to 0	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.7 Reset Control Register (RSTCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FCETX STP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SWRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	FCETXSTP	0	RW	Force Tx Stop Mode When this value is “1”, force to change to Transmit mode and StopState by ForceTxStopmode on all PPI Data Lane. Set “1” when SWRST=0 is prohibited.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SWRST	0	RW	Request of Software Reset Set to “1” to request for Software Reset. After the request of Software Reset, wait to confirm the completion of the Reset procedure by RSTSR.SWRSTx=1. And set to “0” after that.

34.3.1.8 Reset Status Register (RSTSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL0DIR	—	—	—	DLSTPST				—	—	—	SWRST V1	SWRST IB	SWRST APB	SWRST LP	SWRST HS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	DL0DIR	0	R	Direction (Tx or Rx) of PPI Data Lane0 0: PHY is in Tx Mode 1: PHY is in Rx Mode This field is copy of PLSR.DL0DIR
14 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	DLSTPST	All 0	R	Status of Stopstate on Data Lane bit 3: Stopstate of Data Lane3 bit 2: Stopstate of Data Lane2 bit 1: Stopstate of Data Lane1 bit 0: Stopstate of Data Lane0 This field is copy of PLSR.DLSTPST
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	SWRSTV1	0	R	Status of Software Reset by RSTCR.SWRST at MIPI_DSI_VCLK 0: Not on Reset procedure at MIPI_DSI_VCLK 1: Now on Reset procedure at MIPI_DSI_VCLK
3	SWRSTIB	0	R	Status of Software Reset by RSTCR.SWRST at MIPI_DSI_ACLK 0: Not on Reset procedure at MIPI_DSI_ACLK 1: Now on Reset procedure at MIPI_DSI_ACLK
2	SWRSTAPB	0	R	Status of Software Reset by RSTCR.SWRST at MIPI_DSI_PCLK 0: Not on Reset procedure at MIPI_DSI_PCLK 1: Now on Reset procedure at MIPI_DSI_PCLK
1	SWRSTLP	0	R	Status of Software Reset by RSTCR.SWRST at MIPI_DSI_LPCLK 0: Not on Reset procedure at MIPI_DSI_LPCLK 1: Now on Reset procedure at MIPI_DSI_LPCLK
0	SWRSTHS	0	R	Status of Software Reset by RSTCR.SWRST at hsclk 0: Not on Reset procedure at hsclk 1: Now on Reset procedure at hsclk

34.3.1.9 DSI Set Register (DSISETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EOTPEN	TEIDIR	SCREN	—	—	—	—	—	CRCEN				—	—	—	ECCEN
Initial Value	1	0	0	0	0	0	0	0	1	1	1	1	0	0	0	1
R/W	RW	RW	RW	R	R	R	R	R	RW	RW	RW	RW	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRPSZ															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	EOTPEN	0	RW	HS Transfer EoTp Enable 0: Transfer EoTp disable 1: Transfer EoTp enable EoTp is always disabled in LP Transfer.
30	TEIDIR	0	RW	Tearing Effect Input Direction External Tearing Effect Setting 0: rise edge detect 1: fall edge detect
29	SCREN	0	RW	Scramble Enable 0: Scrambling data is disabled 1: Scrambling data is enabled This bit should not set to “1”, if peer-device does not support data scrambling function.
28 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 20	CRCEN	H’F	RW	Force Tx Stop Mode When this value is “1”, force to change to Transmit mode and StopState by ForceTxStopmode on all PPI Data Lane. Set “1” when SWRST=0 is prohibited.
19 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	ECCEN	0	RW	Enable Received ECC check 0: Received ECC check is disabled 1: Received ECC check is enabled
15 to 0	MRPSZ	H’0001	RW	Maximum Return Packet Size If returned long packet’s WC is over this value, the DSI-Tx Module asserts RXSR.MAXRPSZERR error. In that case data will not save to memory space. Setting 0 for this field is prohibited. Set this field equal or more than peripheral setting.

34.3.1.10 Receive Buffer Size Register (RXBUFSZR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	RXBUFSZ				—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 20	RXBUFSZ	H’0	R	Receive Buffer Size H’0: 128B
19 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.11 Tx Packet Payload Data 0 Register (TXPPD0R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	RW	Tx Packet Payload Data 0 When sequence operation Long packet is set (SQCHxDSCyAR.FMT=1) and Payload Data Access use Packet Payload Data register (SQCHxDSCyBR.DTSEL =H’0) , the values in this register is used for Long Packet Payload. [7:0]: Data 0 [15:8]: Data 1 [23:16]: Data 2 [31:24]: Data 3

34.3.1.12 Tx Packet Payload Data 1 Register (TXPPD1R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	RW	Tx Packet Payload Data 1 When sequence operation Long packet is set (SQCHxDSCyAR.FMT=1) and Payload Data Access use Packet Payload Data register (SQCHxDSCyBR.DTSEL =H'0) , the values in this register is used for Long Packet Payload. [7:0]: Data 4 [15:8]: Data 5 [23:16]: Data 6 [31:24]: Data 7

34.3.1.13 Tx Packet Payload Data 2 Register (TXPPD2R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	RW	Tx Packet Payload Data 2 When sequence operation Long packet is set (SQCHxDSCyAR.FMT=1) and Payload Data Access use Packet Payload Data register (SQCHxDSCyBR.DTSEL =H’0) , the values in this register is used for Long Packet Payload. [7:0]: Data 8 [15:8]: Data 9 [23:16]: Data 10 [31:24]: Data 11

34.3.1.14 Tx Packet Payload Data 3 Register (TXPPD3R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	RW	Tx Packet Payload Data 3 When sequence operation Long packet is set (SQCHxDSCyAR.FMT=1) and Payload Data Access use Packet Payload Data register (SQCHxDSCyBR.DTSEL =H’0) , the values in this register is used for Long Packet Payload. [7:0]: Data 12 [15:8]: Data 13 [23:16]: Data 14 [31:24]: Data 15

34.3.1.15 Rx Status Register (RXSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RXAKE	—	ECCERR1B	—	MAXRPSZERR	NORETERR	PRESPTOERR	RXOVFERR	IBERR	CRCERR	WCERR	—	UEXPKTERR	ECCERR	MLFERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXTTEDET	RXACK	RXTE	RXRTRG	RXUK5TRG	RXEOTP	—	RXRESP	—	—	—	—	—	TATO	LRXHTO	BTAREQEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	RXAKE	0	R	Rx Acknowledge and Error Report When this value is “1”, the Acknowledge and Error Report Packet is received. Write “1” to RXSCR.RXAKE register to clear this field.
29	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	ECCERR1B	0	R	ECC Error (1bit) When this value is “1”, the Rx ECC Error (1bit) is detected. Write “1” to RXSCR.ECCERR1B register to clear this field.
27	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	MAXRPSZERR	0	R	Maximum Return Packet Size Error When this value is “1”, the WC value of Rx Long Packet is larger than DSISETR.MRPSZ. Write “1” to RXSCR.MAXRPSZERR register to clear this field.
25	NORETERR	0	R	Nothing Return Error When this value is “1”, no trigger or packet is returned in BTA period. Write “1” to RXSCR.NORETERR register to clear this field.
24	PRESPTOERR	0	R	Peripheral Response Timeout Error When this value is “1”, the Rx Peripheral Response Timeout is detected after changing to Rx mode along with Bus Turn-Around and within the time specified by PRESPTOBTASETR.PRESPTOBTA, PRESPTOLPSETR.PRESPTOLPR, PRESPTOLPSETR.PRESPTOLPW, PRESPTOLPSETR.PRESPTOHSR or PRESPTOLPSETR.PRESPTOHSW. The value is selected from kind of BTA. Write “1” to RXSCR.PRESPTOERR register to clear this field.
23	RXOVFERR	0	R	Receive Buffer Overflow Error When this value is “1”, the Buffer Overflow Error is detected at Rx Long Packet. Write “1” to RXSCR.RXOVFERR register to clear this field.
22	IBERR	0	R	Internal Bus Error When this value is “1”, the Internal Bus Write had failed response. Write “1” to RXSCR.IBERR register to clear this field.
21	CRCERR	0	R	CRC Error When this value is “1”, the Rx CRC Error is detected. Write “1” to RXSCR.CRCERR register to clear this field.
20	WCERR	0	R	Word Count Error When this value is “1”, the Rx Packet’s actual WC length is shorter than packet header’s WC. Write “1” to RXSCR.WCERR register to clear this field.

Bit	Bit Name	Initial Value	R/W	Description
19	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	UEXPKTERR	0	R	Unexpected Packet Error When this value is “1”, the unexpected Packet is received. “Unexpected Packet” means unexpected DT for receive or unexpected response for receive. Write “1” to RXSCR.UEXPKTERR register to clear this field.
17	ECCERR	0	R	ECC Error (over 1bit) When this value is “1”, the Rx ECC Error (over 1bit) is detected. Write “1” to RXSCR.ECCERR register to clear this field.
16	MLFERR	0	R	Malform Error When this value is “1”, the Rx Packet under 4 Byte length is received. Write “1” to RXSCR.MLFERR register to clear this field.
15	EXTTEDET	0	R	External Tearing Effect Detect When this value is “1”, External Tearing Effect is detected. Write “1” to RXSCR.EXTTEDET register to clear this field.
14	RXACK	0	R	Rx ACK Trigger When this value is “1”, the ACK trigger is received. Write “1” to RXSCR.RXACK register to clear this field.
13	RXTE	0	R	Rx Tearing Effect Trigger When this value is “1”, Tearing Effect Trigger is received. Write “1” to RXSCR.RXTE register to clear this field.
12	RXRTRG	0	R	Rx Reset Trigger When this value is “1”, Reset Trigger is received. Normally, the DSI-Tx Module does not receive Reset Trigger. Write “1” to RXSCR.RXRTRG register to clear this field.
11	RXUK5TRG	0	R	Rx Unknown-5 Trigger When this value is “1”, Unknown-5 Trigger is received. Normally, the DSI-Tx Module does not receive Unknown-5 Trigger. Write “1” to RXSCR.RXUK5TRG register to clear this field.
10	RXEOTP	0	R	Rx EoTp When this value is “1”, the EoTp is received. Write “1” to RXSCR.RXEOTP register to clear this field.
9	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	RXRESP	0	R	Rx Response Packet When this value is “1”, the Response Packet is received. Write “1” to RXSCR.RXRESP register to clear this field.
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	TATO	0	R	Turnaround Acknowledge Timeout When this value is “1”, the Turnaround Acknowledge Timeout (TA_TO) is detected. Write “1” to RXSCR.TATO register to clear this field.
1	LRXHTO	0	R	LP-RX Host Processor Timeout When this value is “1”, the LP-RX Host Processor Timeout (LRX-H_TO) is detected. Write “1” to RXSCR.LRXHTO register to clear this field.
0	BTAREQEND	0	R	BTA Request End When this value is “1”, the completion of Transmission request includes Rx is detected. This bit is set if all BTA requests are finished including error. Status of BTA request are indicated by RXSR[31:1]. Write “1” to RXSCR.BTAREQEND register to clear this field.

34.3.1.16 Rx Status Clear Register (RXSCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RXAKE	—	ECCERR1B	—	MAXRPSZERR	NORETERR	PRESPTOERR	RXOVFERR	IBERR	CRCERR	WCERR	—	UEXPKTERR	ECCERR	MLFERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R	R0W	R	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXTTEDET	RXACK	RXTE	RXRTRG	RXUK5TRG	RXEOTP	—	RXRESP	—	—	—	—	—	TATO	LRXHTO	BTAREQEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W	R0W	R0W	R0W	R0W	R0W	R	R0W	R	R	R	R	R	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	RXAKE	0	R0W	RXSR.RXAKE Clear Set to “1” to clear the RXSR.RXAKE Writing “0” has no effect.
29	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	ECCERR1B	0	R0W	RXSR.ECCERR1B Clear Set to “1” to clear the RXSR.ECCERR1B Writing “0” has no effect.
27	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	MAXRPSZERR	0	R0W	RXSR.MAXRPSZERR Clear Set to “1” to clear the RXSR.MAXRPSZERR Writing “0” has no effect.
25	NORETERR	0	R0W	RXSR.NORETERR Clear Set to “1” to clear the RXSR.NORETERR Writing “0” has no effect.
24	PRESPTOERR	0	R0W	RXSR.PRESPTOERR Clear Set to “1” to clear the RXSR.PRESPTOERR Writing “0” has no effect.
23	RXOVFERR	0	R0W	RXSR.RXOVFERR Clear Set to “1” to clear the RXSR.RXOVFERR Writing “0” has no effect.
22	IBERR	0	R0W	RXSR.IBERR Clear Set to “1” to clear the RXSR.IBERR Writing “0” has no effect.
21	CRCERR	0	R0W	RXSR.CRCERR Clear Set to “1” to clear the RXSR.CRCERR Writing “0” has no effect.
20	WCERR	0	R0W	RXSR.WCERR Clear Set to “1” to clear the RXSR.WCERR Writing “0” has no effect.
19	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18	UEXPKTERR	0	R0W	RXSR.UEXPKTERR Clear Set to “1” to clear the RXSR.UEXPKTERR Writing “0” has no effect.
17	ECCERR	0	R0W	RXSR.ECCERR Clear Set to “1” to clear the RXSR.ECCERR Writing “0” has no effect.
16	MLFERR	0	R0W	RXSR.MLFERR Clear Set to “1” to clear the RXSR.MLFERR Writing “0” has no effect.
15	EXTTEDET	0	R0W	RXSR.EXTTTEDET Clear Set to “1” to clear the RXSR.EXTTTEDET Writing “0” has no effect.
14	RXACK	0	R0W	RXSR.RXACK Clear Set to “1” to clear the RXSR.RXACK Writing “0” has no effect.
13	RXTE	0	R0W	RXSR.RXTE Clear Set to “1” to clear the RXSR.RXTE Writing “0” has no effect.
12	RXRTRG	0	R0W	RXSR.RXRTRG Clear Set to “1” to clear the RXSR.RXRTRG Writing “0” has no effect.
11	RXUK5TRG	0	R0W	RXSR.RXUK5TRG Clear Set to “1” to clear the RXSR.RXUK5TRG Writing “0” has no effect.
10	RXEOTP	0	R0W	RXSR.RXEOTP Clear Set to “1” to clear the RXSR.RXEOTP Writing “0” has no effect.
9	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	RXRESP	0	R0W	RXSR.RXRESP Clear Set to “1” to clear the RXSR.RXRESP Writing “0” has no effect.
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	TATO	0	R0W	RXSR.TATO Clear Set to “1” to clear the RXSR.TATO. Writing “0” has no effect.
1	LRXHTO	0	R0W	RXSR.LRXHTO Clear Set to “1” to clear the RXSR.LRXHTO. Writing “0” has no effect.
0	BTAREQEND	0	R0W	RXSR.BTAREQEND Clear Set to “1” to clear the RXSR.BTAREQEND Writing “0” has no effect.

34.3.1.17 Rx Interrupt Enable Register (RXIER)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RXAKE	—	ECCERR1B	—	MAXRPSZERR	NORETERR	PRESPTOERR	RXOVFERR	IBERR	CRCERR	WCERR	—	UEXPKTERR	ECCERR	MLFERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	R	RW	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXTTEDET	RXACK	RXTE	RXRTRG	RXUK5TRG	RXEOTP	—	RXRESP	—	—	—	—	—	TATO	LRXHTO	BTAREQEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	R	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	RXAKE	0	RW	Interrupt enable for RXSR.RXAKE 0: Disable interrupt of dsi_int_rcv when RXSR.RXAKE =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXAKE =1
29	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	ECCERR1B	0	RW	Interrupt enable for RXSR.ECCERR1B 0: Disable interrupt of dsi_int_rcv when RXSR.ECCERR1B=1 1: Enable interrupt of dsi_int_rcv when RXSR.ECCERR1B=1
27	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	MAXRPSZERR	0	RW	Interrupt enable for RXSR.MAXRPSZERR 0: Disable interrupt of dsi_int_rcv when RXSR.MAXRPSZERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.MAXRPSZERR =1
25	NORETERR	0	RW	Interrupt enable for RXSR.NORETERR 0: Disable interrupt of dsi_int_rcv when RXSR.NORETERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.NORETERR =1
24	PRESPTOERR	0	RW	Interrupt enable for RXSR.PRESPTOERR 0: Disable interrupt of dsi_int_rcv when RXSR.PRESPTOERR=1 1: Enable interrupt of dsi_int_rcv when RXSR.PRESPTOERR=1
23	RXOVFERR	0	RW	Interrupt enable for RXSR.RXOVFERR 0: Disable interrupt of dsi_int_rcv when RXSR.RXOVFERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXOVFERR =1
22	IBERR	0	RW	Interrupt enable for RXSR.IBERR 0: Disable interrupt of dsi_int_rcv when RXSR.IBERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.IBERR =1
21	CRCERR	0	RW	Interrupt enable for RXSR. CRCERR 0: Disable interrupt of dsi_int_rcv when RXSR.CRCERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.CRCERR =1
20	WCERR	0	RW	Interrupt enable for RXSR. WCERR 0: Disable interrupt of dsi_int_rcv when RXSR.WCERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.WCERR =1
19	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18	UEXPKTERR	0	RW	Interrupt enable for RXSR.UEXPKTERR 0: Disable interrupt of dsi_int_rcv when RXSR.UEXPKTERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.UEXPKTERR =1
17	ECCERR	0	RW	Interrupt enable for RXSR.ECCERR 0: Disable interrupt of dsi_int_rcv when RXSR.ECCERR=1 1: Enable interrupt of dsi_int_rcv when RXSR.ECCERR=1
16	MLFERR	0	RW	Interrupt enable for RXSR.MLFERR 0: Disable interrupt of dsi_int_rcv when RXSR.MLFERR =1 1: Enable interrupt of dsi_int_rcv when RXSR.MLFERR =1
15	EXTTEDET	0	RW	Interrupt enable for RXSR.EXTTEDET 0: Disable interrupt of dsi_int_rcv when RXSR.EXTTEDET =1 1: Enable interrupt of dsi_int_rcv when RXSR.EXTTEDET =1
14	RXACK	0	RW	Interrupt enable for RXSR.RXACK 0: Disable interrupt of dsi_int_rcv when RXSR.RXACK =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXACK =1
13	RXTE	0	RW	Interrupt enable for RXSR.RXTE 0: Disable interrupt of dsi_int_rcv when RXSR.RXTE =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXTE =1
12	RXRTRG	0	RW	Interrupt enable for RXSR.RXRTRG 0: Disable interrupt of dsi_int_rcv when RXSR.RXRTRG =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXRTRG =1
11	RXUK5TRG	0	RW	Interrupt enable for RXSR.RXUK5TRG 0: Disable interrupt of dsi_int_rcv when RXSR.RXUK5TRG =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXUK5TRG =1
10	RXEOTP	0	RW	Interrupt enable for RXSR.RXEOTP 0: Disable interrupt of dsi_int_rcv when RXSR.RXEOTP =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXEOTP =1
9	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	RXRESP	0	RW	Interrupt enable for RXSR.RXRESP 0: Disable interrupt of dsi_int_rcv when RXSR.RXRESP =1 1: Enable interrupt of dsi_int_rcv when RXSR.RXRESP =1
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	TATO	0	RW	Interrupt enable for RXSR.TATO 0: Disable interrupt of dsi_int_rcv when RXSR.TATO =1 1: Enable interrupt of dsi_int_rcv when RXSR.TATO =1
1	LRXHTO	0	RW	Interrupt enable for RXSR.LRXHTO 0: Disable interrupt of dsi_int_rcv when RXSR.LRXHTO =1 1: Enable interrupt of dsi_int_rcv when RXSR.LRXHTO =1
0	BTAREQEND	0	RW	Interrupt enable for RXSR.BTAREQEND 0: Disable interrupt of dsi_int_rcv when RXSR.BTAREQEND =1 1: Enable interrupt of dsi_int_rcv when RXSR.BTAREQEND =1

34.3.1.18 Peripheral Response Timeout BTA Set Register (PRESPTOBTASETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PRESPTOBTA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRESPTOBTA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PRESPTOBTA	All 0	RW	Peripheral Response Timeout (Bus Turn Around) Count Set Timeout value to measure the interval between Start of Rx mode and Start of Packet Receive along with Bus Turn-Around with SQCHxDSCyAR.BTA= H'3. The limit Time for Timeout is calculated by formula of PRESPTOBTA * (period of MIPI_DSI_LPCLK). When RXSR.PRESPTOERR is "1", the Timeout is detected. Notice, the Timeout is not detected when PRESPTOBTA=0. Since counting clock is asynchronous with Packet Receive, there is some jitter in the Timeout.

34.3.1.19 Peripheral Response Timeout LP Set Register (PRESPTOLPSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PRESPTOLPR															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRESPTOLPW															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PRESPTOLPR	All 0	RW	Peripheral Response Timeout (LPDT READ Request) Count Set Timeout value to measure the interval between Start of Rx mode and Start of Packet Receive along with Bus Turn-Around with SQCHxDSCyAR.BTA=H’2 and SQCHxDSCyAR.SPD=1b. The limit Time for Timeout is calculated by formula of PRESPTOLPR * (period of MIPI_DSI_LPCLK). When RXSR.PRESPTOERR is “1”, the Timeout is detected. Notice, the Timeout is not detected when PRESPTOLPR=0. Since counting clock is asynchronous with Packet Receive, there is some jitter in the Timeout.
15 to 0	PRESPTOLPW	All 0	RW	Peripheral Response Timeout (LPDT WRITE Request) Count Set Timeout value to measure the interval between Start of Rx mode and Start of Packet Receive along with Bus Turn-Around with SQCHxDSCyAR.BTA=01b and SQCHxDSCyAR.SPD=1b. The limit Time for Timeout is calculated by formula of PRESPTOLPW * (period of MIPI_DSI_LPCLK). When RXSR.PRESPTOERR is “1”, the Timeout is detected. Notice, the Timeout is not detected when PRESPTOLPW=0. Since counting clock is asynchronous with Packet Receive, there is some jitter in the Timeout.

34.3.1.20 Peripheral Response Timeout HS Set Register (PRESPTOHSSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PRESPTOHSR															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRESPTOHSW															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PRESPTOHSR	All 0	RW	Peripheral Response Timeout (HS READ Request) Count Set Timeout value to measure the interval between Start of Rx mode and Start of Packet Receive along with Bus Turn-Around with SQCHxDSCyAR.BTA=H’2 and SQCHxDSCyAR.SPD=0b. The limit Time for Timeout is calculated by formula of PRESPTOHSR * (period of MIPI_DSI_LPCLK). When RXSR.PRESPTOERR is “1”, the Timeout is detected. Notice, the Timeout is not detected when PRESPTOHSR=0. Since counting clock is asynchronous with Packet Receive, there is some jitter in the Timeout.
15 to 0	PRESPTOHSW	All 0	RW	Peripheral Response Timeout (HS WRITE Request) Count Set Timeout value to measure the interval between Start of Rx mode and Start of Packet Receive along with Bus Turn-Around with SQCHxDSCyAR.BTA=H’1 and SQCHxDSCyAR.SPD=0b. The limit Time for Timeout is calculated by formula of PRESPTOHSW * (period of MIPI_DSI_LPCLK). When RXSR.PRESPTOERR is “1”, the Timeout is detected. Notice, the Timeout is not detected when PRESPTOHSW=0. Since counting clock is asynchronous with Packet Receive, there is some jitter in the Timeout.

34.3.1.21 Acknowledge and Error Report Packet Parameter Latest Info Register (AKEPLATIR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	VC3	VC2	VC1	VC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRRPTLAT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	−	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	VC3	0	R	Virtual Channel Identifier 3 Latest This field shows Virtual channel identifier of received Acknowledge and Error Report Packet. 0: Virtual channel identifier is not 3 1: Virtual channel identifier is 3 This field shows latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
18	VC2	0	R	Virtual Channel Identifier 2 Latest This field shows Virtual channel identifier of received Acknowledge and Error Report Packet. 0: Virtual channel identifier is not 2 1: Virtual channel identifier is 2 This field shows latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
17	VC1	0	R	Virtual Channel Identifier 1 Latest This field shows Virtual channel identifier of received Acknowledge and Error Report Packet. 0: Virtual channel identifier is not 1 1: Virtual channel identifier is 1 This field shows latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
16	VC0	0	R	Virtual Channel Identifier 0 Latest This field shows Virtual channel identifier of received Acknowledge and Error Report Packet. 0: Virtual channel identifier is not 0 1: Virtual channel identifier is 0 This field shows latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
15 to 0	ERRRPTLAT	All 0	R	Error Report Latest This field shows Error Report of received Acknowledge and Error Report Packet. This field shows latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.

34.3.1.22 Acknowledge and Error Report Packet Parameter Accumulate Status Register (AKEPACMSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	VC3	VC2	VC1	VC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRRPTACM															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	VC3	0	R	Virtual Channel Identifier 3 Accumulate This field is set to “1”, when an Acknowledge and Error Report Packet whose Virtual channel identifier = 3 is received. This field shows accumulated information. This bit is cleared by AKEPSCR.VC3.
18	VC2	0	R	Virtual Channel Identifier 2 Accumulate This field is set to “1”, when an Acknowledge and Error Report Packet whose Virtual channel identifier = 2 is received. This field shows accumulated information. This bit is cleared by AKEPSCR.VC2.
17	VC1	0	R	Virtual Channel Identifier 1 Accumulate This field is set to “1”, when an Acknowledge and Error Report Packet whose Virtual channel identifier = 1 is received. This field shows accumulated information. This bit is cleared by AKEPSCR.VC1.
16	VC0	0	R	Virtual Channel Identifier 0 Accumulate This field is set to “1”, when an Acknowledge and Error Report Packet whose Virtual channel identifier = 0 is received. This field shows accumulated information. This bit is cleared by AKEPSCR.VC0.
15 to 0	ERRRPTACM	All 0	R	Error Report Accumulate Each bit of this field is set to “1”, when an Acknowledge and Error Report Packet whose each Error Report bit = 1 is received. This field shows accumulated information. Each bit is cleared by corresponding AKESCR.ERRRPTACM.

34.3.1.23 Acknowledge and Error Report Packet Parameter Status Clear Register (AKEPSCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	VC3	VC2	VC1	VC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRRPTACM															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	VC3	0	R0W	AKEPACMSR.VC3 Clear Set to “1” to clear AKEPACMSR.VC3. Writing “0” has no effect.
18	VC2	0	R0W	AKEPACMSR.VC2 Clear Set to “1” to clear AKEPACMSR.VC2. Writing “0” has no effect.This field shows accumulated information. This bit is cleared by AKEPSCR.VC2.
17	VC1	0	R0W	AKEPACMSR.VC1 Clear Set to “1” to clear AKEPACMSR.VC1. Writing “0” has no effect.
16	VC0	0	R0W	AKEPACMSR.VC0 Clear Set to “1” to clear AKEPACMSR.VC0. Writing “0” has no effect.
15 to 0	ERRRPTACM	All 0	R0W	Error Report Accumulate Clear Set to “1” to clear each AKEPACMSR.ERRRPTACM. Writing “0” has no effect.

34.3.1.24 Rx Result Saved Status Register (RXRSSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SLT3VLD	SLT2VLD	SLT1VLD	SLT0VLD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	SLT3VLD	0	R	Slot 3 Valid Response packet is received, or information is saved to RXRSS3R. Write 1 to RXRSSCR.SLT3C to clear this bit. This bit and RXRSS3R are set if next receive action caused with SQCHxDSCyCR.ACTCODE = 03 again.
2	SLT2VLD	0	R	Slot 2 Valid Response packet is received, or information is saved to RXRSS2R. Write 1 to RXRSSCR.SLT2C to clear this bit. This bit and RXRSS2R are set if next receive action caused with SQCHxDSCyCR.ACTCODE = 02 again.
1	SLT1VLD	0	R	Slot 1 Valid Response packet is received, or information is saved to RXRSS1R. Write 1 to RXRSSCR.SLT1C to clear this bit. This bit and RXRSS1R are set if next receive action caused with SQCHxDSCyCR.ACTCODE = 01 again.
0	SLT0VLD	0	R	Slot 0 Valid Response packet is received, or information is saved to RXRSS0R. Write 1 to RXRSSCR.SLT0C to clear this bit. This bit and RXRSS0R are set if next receive action caused with SQCHxDSCyCR.ACTCODE = 00 again.

34.3.1.25 Rx Result Saved Status Clear Register (RXRSSCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SLT3C	SLT2C	SLT1C	SLT0C
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	SLT3C	0	R0W	RXRSSR.SLT3VLD Clear Write 1 to clear RXRSSR.SLT3VLD. Writing “0” has no effect.
2	SLT2C	0	R0W	RXRSSR.SLT2VLD Clear Write 1 to clear RXRSSR.SLT2VLD. Writing “0” has no effect.
1	SLT1C	0	R0W	RXRSSR.SLT1VLD Clear Write 1 to clear RXRSSR.SLT1VLD. Writing “0” has no effect.
0	SLT0C	0	R0W	RXRSSR.SLT0VLD Clear Write 1 to clear RXRSSR.SLT0VLD. Writing “0” has no effect.

34.3.1.26 Rx Result Info Overwrite Status Register (RXRINFOOWSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SLT3INFOOW	SLT2INFOOW	SLT1INFOOW	SLT0INFOOW
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
24	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 4	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
3	SLT3INFOOW	0	R	Slot 3 Information Overwrite This bit is copy of RXRSS3R.INFOOW.
2	SLT2INFOOW	0	R	Slot 2 Information Overwrite This bit is copy of RXRSS2R.INFOOW.
1	SLT1INFOOW	0	R	Slot 1 Information Overwrite This bit is copy of RXRSS1R.INFOOW.
0	SLT0INFOOW	0	R	Slot 0 Information Overwrite This bit is copy of RXRSS0R.INFOOW.

34.3.1.27 Rx Result Info Overwrite Status Clear Register (RXRINFOOWSCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SLT3INFOOWC	SLT2INFOOWC	SLT1INFOOWC	SLT0INFOOWC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	SLT3INFOOWC	0	R0W	RXRSS3R.INFOOW Clear Write 1 to clear RXRSS3R.INFOOW. Writing “0” has no effect. To prevent missing of RXRSS3R.INFOOW, clear RXRSS3R.INFOOW before clearing RXRSSR.SLT3VLD.
2	SLT2INFOOWC	0	R0W	RXRSS2R.INFOOW Clear Write 1 to clear RXRSS2R.INFOOW. Writing “0” has no effect. To prevent missing of RXRSS2R.INFOOW, clear RXRSS2R.INFOOW before clearing RXRSSR.SLT2VLD.
1	SLT1INFOOWC	0	R0W	RXRSS1R.INFOOW Clear Write 1 to clear RXRSS1R.INFOOW. Writing “0” has no effect. To prevent missing of RXRSS1R.INFOOW, clear RXRSS1R.INFOOW before clearing RXRSSR.SLT1VLD.
0	SLT0INFOOWC	0	R0W	RXRSS0R.INFOOW Clear Write 1 to clear RXRSS0R.INFOOW. Writing “0” has no effect. To prevent missing of RXRSS0R.INFOOW, clear RXRSS0R.INFOOW before clearing RXRSSR.SLT0VLD.

34.3.1.28 Rx Result Save Slot x Register (RXRSSxR)

x = 0, 1, 2, 3.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INFOOW	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	RXSUC	FMT	VC		DT					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA1								DATA0							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	INFOOW	0	R	Information Overwrite This bit is set if information of RXRSSxR are updated when RXRSSR.SLTxVLD is 1. Write 1 to RXRINFOOWSCR.SLTxINFOOWC to clear this bit.
30	RXAKE	0	R	Receive Acknowledge and Error Report Packet RXSR.RXAKE is also reported.
29	RXCORERR	0	R	Receive Correctable Error Received packet has Correctable Error. RXSR.ECCERR1B is also reported.
28	RXPKTDFAIL	0	R	Receive Packet Data Fail Packet Header Saved Correctly but Data did not save correctly. One or more of the following bits are also reported. RXSR.CRCERR, RXSR.WCERR, RXSR.MAXRPSZERR, RXSR.UEXPKTERR, RXSR.RXOVFERR or RXSR.IBERR. When both of this bit and RXRESP are asserted, it means redundant packet received in BTA.
27	RXFAIL	0	R	Receive Fail Expected receive did not done. One or more of the following bits are also reported. RXSR.PRESPTOERR, RXSR.ECCERR, RXSR.MLFERR or RXSR.NORETERR.
26	RXFATALERR	0	R	Fatal Error Fatal timeout occurred during BTA. One or more of the following bits are also reported. FERRSR.TATO or FERRSR.LRXHTO.
25	RXSUC	0	R	Receive Success When this value is “1”, the Response Packet or ACK trigger is received. One or more of the following bits are also reported. RXSR.RXRESP or RXSR.RXACK.
24	FMT	0	R	Packet format of Rx Packet Header 0: Short packet 1: Long packet This field is valid when RXPKTDFAIL=1 or (RXSUC=1 and DT!=0).
23 to 22	VC	All 0	R	Identifier of virtual channel of Rx Packet Header This field is valid when (RXSUC=1 and DT!=0).
21 to 16	DT	All 0	R	Data type of Rx Packet Header H’00: ACK trigger received H’01 to H’3F: Data type This field is valid when RXSUC=1.

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	DATA1	All 0	R	Data1 of Rx Packet Header When Rx Packet format is Long Packet, this value is upper 8 bits of the word count. This field is valid when (RXSUC=1 and DT!=0).
7 to 0	DATA0	All 0	R	Data0 of Rx Packet Header When Rx Packet format is Long Packet, this value is lower 8 bits of the word count. This field is valid when (RXSUC=1 and DT!=0).

34.3.1.29 Rx Packet Payload Data 0 Register (RXPPD0R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	R	Rx Packet Payload Data 0 In the case of long packet receiving Operation with SQCHxDSCyBR.DTSEL = H'0, received packet payload are saved. [7:0]: Data 0 [15:8]: Data 1 [23:16]: Data 2 [31:24]: Data 3

34.3.1.30 Rx Packet Payload Data 1 Register (RXPPD1R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	R	Rx Packet Payload Data 1 In the case of long packet receiving Operation with SQCHxDSCyBR.DTSEL = H'0. Received packet payload are saved. [7:0]: Data 4 [15:8]: Data 5 [23:16]: Data 6 [31:24]: Data 7

34.3.1.31 Rx Packet Payload Data 2 Register (RXPPD2R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	R	Rx Packet Payload Data 2 In the case of long packet receiving Operation with SQCHxDSCyBR.DTSEL = H'0. Received packet payload are saved. [7:0]: Data 8 [15:8]: Data 9 [23:16]: Data 10 [31:24]: Data 11

34.3.1.32 Rx Packet Payload Data 3 Register (RXPPD3R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA	All 0	R	Rx Packet Payload Data 3 In the case of long packet receiving Operation with SQCHxDSCyBR.DTSEL = H'0. Received packet payload are saved. [7:0]: Data 12 [15:8]: Data 13 [23:16]: Data 14 [31:24]: Data 15

34.3.1.33 HSTX Timeout Set Register (HSTXTOSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HTXTO															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HTXTO															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	HTXTO	All 0	RW	HS TX Timeout Count (HTX_TO) The limit time for timeout is calculated by formula of HTXTO * (period of highspeed serial UI * 32). When FERRSR.HTXTO is “1”, the timeout is detected. Notice, the timeout is not detected when HTXTO=0.

34.3.1.34 LRX-H Timeout Set Register (LRXHTOSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LRXHTO															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LRXHTO															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LRXHTO	All 0	RW	LP-RX Host Processor Timeout (LRX-H_TO) The limit time for timeout is calculated by formula of LRXHTO * (period of MIPI_DSI_LPCLK). When FERRSR.LRXHTO is “1”, the timeout is detected. Notice, the timeout is not detected when LRXHTO=0.

34.3.1.35 TA Timeout Set Register (TATOSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TATO															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TATO															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TATO	All 0	RW	Turnaround Acknowledge Timeout (TA_TO) The limit time for timeout is calculated by formula of TATO * (period of MIPI_DSI_LPCLK). When FERRSR.TATO is “1”, the timeout is detected. Notice, the timeout is not detected when TATO=0.

34.3.1.36 Fatal Error Status Register (FERRSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	ERRCLP1S	ERRCLP0S	—	—	—	—	—	—	ERRCLP1	ERRCLP0	ERRCTRL	ERRSYNESC	ERRESC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TATO	LRXHTO	HTXTO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	ERRCLP1S	0	R	Status of Contention Error of LP1 (ErrContentionLP1)
27	ERRCLP0S	0	R	Status of Contention Error of LP0 (ErrContentionLP0)
26 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	ERRCLP1	0	R	Contention Error of LP1 (ErrContentionLP1) When this value is “1”, the Contention Error of LP1 (ErrContentionLP1) is detected on PPI Data Lane0. Write “1” to FERRSCR.ERRCLP1 register to clear this field.
19	ERRCLP0	0	R	Contention Error of LP0 (ErrContentionLP0) When this value is “1”, the Contention Error of LP0 (ErrContentionLP0) is detected on PPI Data Lane0. Write “1” to FERRSCR.ERRCLP0 register to clear this field.
18	ERRCTRL	0	R	Control Error (ErrControl) When this value is “1”, the Control Error (ErrControl) is detected on PPI Data Lane0. Write “1” to FERRSCR.ERRCTRL register to clear this field.
17	ERRSYNESC	0	R	Sync Error of LPDT (ErrSyncEsc) When this value is “1”, the Sync Error of LPDT (ErrSyncEsc) is detected on PPI Data Lane0. Write “1” to FERRSCR.ERRSYNESC register to clear this field.
16	ERRESC	0	R	Escape mode Entry Error (ErrEsc) When this value is “1”, the Escape mode Entry Error (ErrEsc) is detected on PPI Data Lane0. Write “1” to FERRSCR.ERRESC register to clear this field.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	TATO	0	R	Turnaround Acknowledge Timeout (TA_TO) When this value is “1”, the Turnaround Acknowledge Timeout (TA_TO) is detected. Write “1” to FERRSCR.TATO register to clear this field.
1	LRXHTO	0	R	LP-RX Host Processor Timeout (LRX-H_TO) When this value is “1”, the LP-RX Host Processor Timeout (LRX-H_TO) is detected. Write “1” to FERRSCR.LRXHTO register to clear this field.
0	HTXTO	0	R	HS TX Timeout (HTX_TO) When this value is “1”, the HS TX Timeout (HTX_TO) is detected. Write “1” to FERRSCR.HTXTO register to clear this field.

34.3.1.37 Fatal Error Status Clear Register (FERRSCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ERRCLP1	ERRCLP0	ERRCTRL	ERRSYNESC	ERRESC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TATO	LRXHTO	HTXTO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	ERRCLP1	0	R0W	FERRSR.ERRCLP1 Clear Set to “1” to clear the FERRSR.ERRCLP1. Writing “0” has no effect.
19	ERRCLP0	0	R0W	FERRSR.ERRCLP0 Clear Set to “1” to clear the FERRSR.ERRCLP0. Writing “0” has no effect.
18	ERRCTRL	0	R0W	FERRSR.ERRCTRL Clear Set to “1” to clear the FERRSR.ERRCTRL. Writing “0” has no effect.
17	ERRSYNESC	0	R0W	FERRSR.ERRSYNESC Clear Set to “1” to clear the FERRSR.ERRSYNESC. Writing “0” has no effect.
16	ERRESC	0	R0W	FERRSR.ERRESC Clear Set to “1” to clear the FERRSR.ERRESC. Writing “0” has no effect.
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	TATO	0	R0W	FERRSR.TATO Clear Set to “1” to clear the FERRSR.TATO. Writing “0” has no effect.
1	LRXHTO	0	R0W	FERRSR.LRXHTO Clear Set to “1” to clear the FERRSR.LRXHTO. Writing “0” has no effect.
0	HTXTO	0	R0W	FERRSR.HTXTO Clear Set to “1” to clear the FERRSR.HTXTO. Writing “0” has no effect.

34.3.1.38 Fatal Error Interrupt Enable Register (FERRIER)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ERRCLP1	ERRCLP0	ERRCTRL	ERRSYNESC	ERRESC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TATO	LRXHTO	HTXTO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

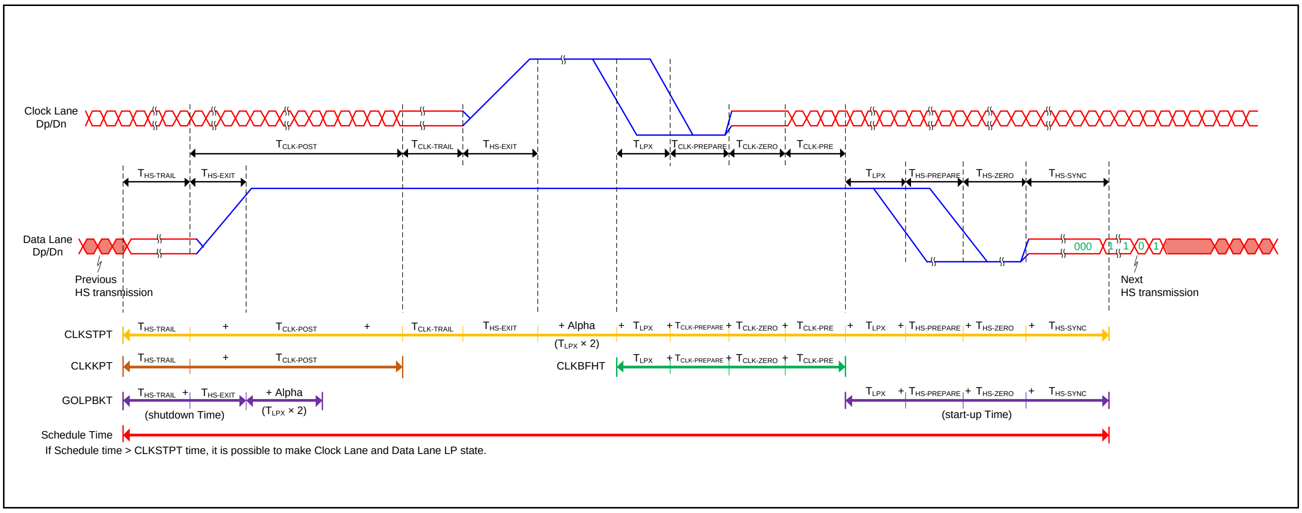
Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	ERRCLP1	0	RW	Interrupt enable for the Contention Error of LP1 on PPI Data Lane0 (ErrContentionLP1) 0: Disable interrupt of dsi_int_ferr when FERRSR.ERRCLP1=1 1: Enable interrupt of dsi_int_ferr when FERRSR.ERRCLP1=1
19	ERRCLP0	0	RW	Interrupt enable for the Contention Error of LP0 on PPI Data Lane0 (ErrContentionLP0) 0: Disable interrupt of dsi_int_ferr when FERRSR.ERRCLP0=1 1: Enable interrupt of dsi_int_ferr when FERRSR.ERRCLP0=1
18	ERRCTRL	0	RW	Interrupt enable for the Control Error on PPI Data Lane0 (ErrControl) 0: Disable interrupt of dsi_int_ferr when FERRSR.ERRCTRL=1 1: Enable interrupt of dsi_int_ferr when FERRSR.ERRCTRL=1
17	ERRSYNESC	0	RW	Interrupt enable for the LPDT Sync Error on PPI Data Lane0 (ErrSyncEsc) 0: Disable interrupt of dsi_int_ferr when FERRSR.ERRSYNESC=1 1: Enable interrupt of dsi_int_ferr when FERRSR.ERRSYNESC=1
16	ERRESC	0	RW	Interrupt enable for the Escape mode Error on PPI Data Lane0 (ErrEsc) 0: Disable interrupt of dsi_int_ferr when FERRSR.ERRESC=1 1: Enable interrupt of dsi_int_ferr when FERRSR.ERRESC=1
15 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	TATO	0	RW	Interrupt enable for Turnaround Acknowledge Timeout (TA_TO) 0: Disable interrupt of dsi_int_ferr when FERRSR.TATO=1 1: Enable interrupt of dsi_int_ferr when FERRSR.TATO=1
1	LRXHTO	0	RW	Interrupt enable for LP-RX Host Processor Timeout (LRX_H_TO) 0: Disable interrupt of dsi_int_ferr when FERRSR.LRXHTO=1 1: Enable interrupt of dsi_int_ferr when FERRSR.LRXHTO=1
0	HTXTO	0	RW	Interrupt enable for HS TX Timeout (HTX_TO) 0: Disable interrupt of dsi_int_ferr when FERRSR.HTXTO=1 1: Enable interrupt of dsi_int_ferr when FERRSR.HTXTO=1

34.3.1.39 Clock Lane Stop Time Set Register (CLSTPTSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CLKKPT								CLKBFHT							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLKSTPT										—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CLKKPT	All 0	RW	Clock Keep Time Period time what HS Clock Lane keep active after HS packet transferred. The time is calculated by formula of CLKKPT * (period of high-speed serial UI * 32). This time corresponds to D-PHY specification's ($T_{\text{HS-TRAIL}}$ + $T_{\text{CLK-POST}}$). Set larger value than calculated. Set appropriate value (other than 0) before HSCLKSETR.HSCLKRUN is set to 1.
23 to 16	CLKBFHT	All 0	RW	Clock Beforehand Time Period time what Clock Lane back to HS beforehand to HS packet transfer. If HS transfers are scheduled in LP, clock lane will start beforehand to HS transfer by this setting. The time is calculated by formula of CLKBFHT * (period of high-speed serial UI * 32). This time corresponds to D-PHY specification's (T_{LPX} + $T_{\text{CLK-PREPARE}}$ + $T_{\text{CLK-ZERO}}$ + $T_{\text{CLK-PRE}}$). Set larger value than calculated. Set appropriate value (other than 0) before HSCLKSETR.HSCLKRUN is set to 1.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 2	CLKSTPT	All 0	RW	Clock Stop Time Period time what HS Clock Lane is changed to LP and back to HS. If data lane will end HS and time of CLKKPT is over, and scheduled time till next HS transfers is more than this setting, clock lane will be changed to LP. This setting is no meaning if HSCLKSETR.HSCLKMODE is 1b or HSCLKSETR.HSCLKRUN is 0b. The time is calculated by formula of CLKSTPT * (period of high-speed serial UI * 32) This time correspond to D-PHY specification's ($T_{\text{HS-TRAIL}}$ + $T_{\text{CLK-POST}}$ + $T_{\text{CLK-TRAIL}}$ + $T_{\text{HS-EXIT}}$ + T_{LPX} *2 + T_{LPX} + $T_{\text{CLK-PREPARE}}$ + $T_{\text{CLK-ZERO}}$ + $T_{\text{CLK-PRE}}$ + T_{LPX} + $T_{\text{HS-PREPARE}}$ + $T_{\text{HS-ZERO}}$ + $T_{\text{HS-SYNC}}$). About each parameter, please refer D-PHY specification and **Figure 34.2**. Set larger value than calculated. Setting 0 to this field is prohibited.
1 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.



If Schedule time > CLKSTPT time, it is possible to make Clock Lane and Data Lane LP state.

Figure 34.2 Switching the Clock Lane between Clock Transmission and Low-Power Mode

34.3.1.40 LP Transition Time Set Register (LPTRNSTSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GOLPBKT									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9 to 0	GOLPBKT	All 0	RW	Go LP and Back Time Period time what HS Data Lanes are changed to LP and back to HS. If scheduled time between HS transfers is more than this setting, data lanes will be changed to LP. The time is calculated by formula of GOLPBKT * (period of high-speed serial 8*UI). This time correspond to D-PHY specification's ($T_{HS\text{-}TRAIL}$ + $T_{HS\text{-}EXIT}$ + T_{LPX} *2 + T_{LPX} + $T_{HS\text{-}PREPARE}$ + $T_{HS\text{-}ZERO}$ + $T_{HS\text{-}SYNC}$). About each parameter, please refer DPHY specification and **Figure 34.3**. The unit of this setting is 8UI, so if the result of this formula contains half-finished than 8UI, round it up. Set the rounded up value plus at least 3.

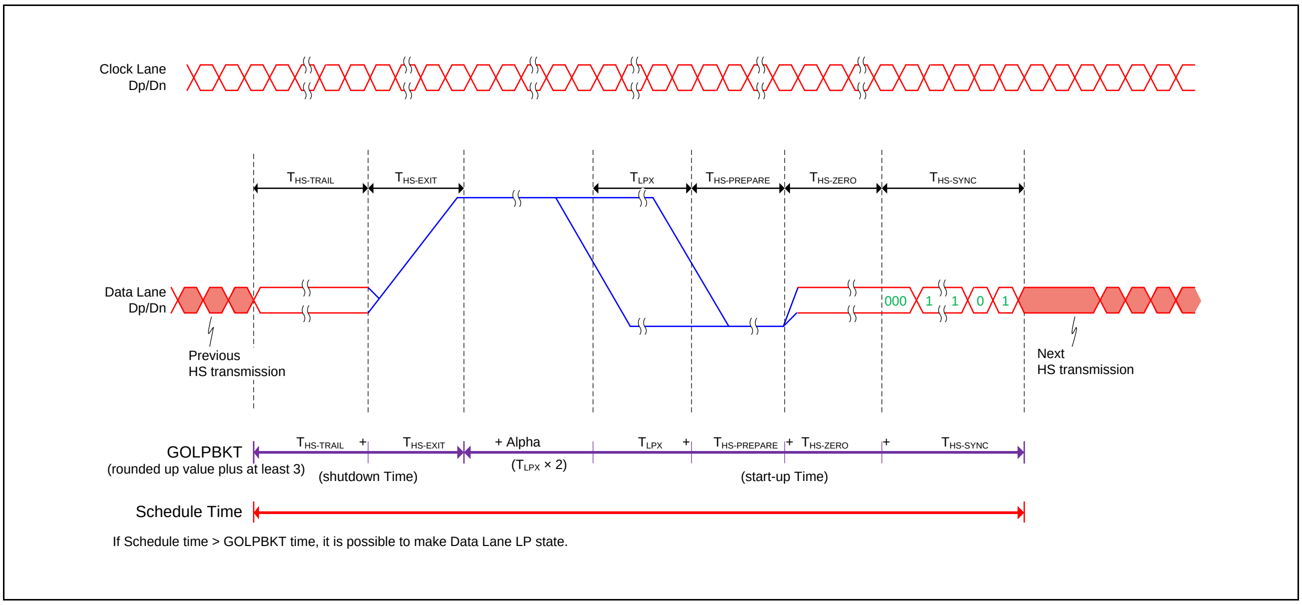


Figure 34.3 Switching the Data Lane between High-Speed Mode and Low-Power Mode

34.3.1.41 Physical Lane Status Register (PLSR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DLFROMULPS	DLTOULPS	CLHS2LP	CLLP2HS	CLFROMULPS	CLTOULPS	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL0DIR	—	DL0TX2RX	DL0RX2TX	DLSTPST				DLULPSACTN				DL0RXULPSESC	DL0RXLPDTESC	CLSTPST	CLULPSACTN
Initial Value	0	0	0	0	*1	*1	*1	*1	1	1	1	1	0	0	*2	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	DLFROMULPS	0	R	From ULPS (Data Lane) When this value is "1", all PPI Data Lanes which enabled by TXSETR.NUMLANEUSE were changed to StopState from ULPS by ULPS Exit. Write "1" to PLSCR.DLFROMULPS register to clear this field.
28	DLTOULPS	0	R	To ULPS (Data Lane) When this value is "1", all PPI Data Lanes which enabled by TXSETR.NUMLANEUSE were changed to ULPS from StopState by ULPS Enter. Write "1" to PLSCR.DLTOULPS register to clear this field.
27	CLHS2LP	0	R	HS to LP (Clock Lane) When this value is "1", the clock lane transitions to LP from HS is detected. Checking this bit during non-continuous clock mode operation is no meaning. So, ignore this bit when HSCLKSETR.HSCLKMODE=0b. Write "1" to PLSCR.CLHS2LP register to clear this field.
26	CLLP2HS	0	R	LP to HS (Clock Lane) When this value is "1", the clock lane transitions to HS from LP is detected. Checking this bit during non-continuous clock mode operation is no meaning. So, ignore this bit when HSCLKSETR.HSCLKMODE=0b. Write "1" to PLSCR.CLLP2HS register to clear this field.
25	CLFROMULPS	0	R	From ULPS (Clock Lane) When this value is "1", the clock lane exits from ULPS is detected. Write "1" to PLSCR.CLFROMULPS register to clear this field.
24	CLTOULPS	0	R	To ULPS (Clock Lane) When this value is "1", the clock lane enters to ULPS is detected. Write "1" to PLSCR.CLTOULPS register to clear this field.
23 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15	DL0DIR	0	R	Direction (Tx or Rx) of Data Lane0 0: PHY is in Tx Mode 1: PHY is in Rx Mode This field shows a status of direction on PPI data lane0 after synchronized to MIPI_DSI_LPCLK. So, this field does not show current status, during MIPI_DSI_LPCLK is not running.
14	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
13	DL0TX2RX	0	R	Tx to Rx on Data Lane0 When this value is “1”, the transition of Direction from 0 (Tx) to 1(Rx) is detected on Data Lane0. Write “1” to PLSCR.DL0TX2RX register to clear this field.
12	DL0RX2TX	0	R	Rx to Tx on Data Lane0 When this value is “1”, the transition of Direction from 1 (Rx) to 0(Tx) is detected on Data Lane0. Write “1” to PLSCR.DL0RX2TX register to clear this field.
11 to 8	DLSTPST	*1	R	Status of Stopstate on Data Lane bit 3: Stopstate of Data Lane3 bit 2: Stopstate of Data Lane2 bit 1: Stopstate of Data Lane1 bit 0: Stopstate of Data Lane0 This field shows a status of Stopstate on PPI data lane after synchronized to MIPI_DSI_LPCLK. So, this field does not show current status, during MIPI_DSI_LPCLK is not running.
7 to 4	DLULPSACTN	H’F	R	Status of UlpsActiveNot on Data Lane bit 3: UlpsActiveNot of Data Lane3 bit 2: UlpsActiveNot of Data Lane2 bit 1: UlpsActiveNot of Data Lane1 bit 0: UlpsActiveNot of Data Lane0 This field shows a status of UlpsActiveNot on PPI data lane after synchronized to MIPI_DSI_LPCLK. So, this field does not show current status, during MIPI_DSI_LPCLK is not running.
3	DL0RXULPSESC	0	R	Status of RxUlpsEsc on Data Lane0 This field shows a status of RxUlpsEsc on PPI data lane0 after synchronized to MIPI_DSI_LPCLK. So, this field does not show current status, during MIPI_DSI_LPCLK is not running.
2	DL0RXLPDTESC	0	R	Status of RxLpdtEsc on Data Lane0 This field shows a status of RxLpdtEsc on PPI data lane0 after synchronized to MIPI_DSI_LPCLK. So, this field does not show current status, during MIPI_DSI_LPCLK is not running.
1	CLSTPST	*2	R	Status of Stopstate on PPI Clock Lane This field shows a status of Stopstate on PPI clock lane after synchronized to MIPI_DSI_LPCLK. So, this field does not show current status, during MIPI_DSI_LPCLK is not running.
0	CLULPSACTN	1	R	Status of UlpsActiveNot on PPI Clock Lane This field shows a status of UlpsActiveNot on PPI clock lane after synchronized to MIPI_DSI_LPCLK. So, this field does not show current status, during MIPI_DSI_LPCLK is not running.

Note 1. This Initial Value depends on ppi_dl0_stopstate, ppi_dl1_stopstate, ppi_dl2_stopstate and ppi_dl3_stopstate.

Note 2. This Initial Value depends on ppi_cl_stopstate.

34.3.1.42 Physical Lane Status Clear Register (PLSCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DLFROMULPS	DLTOULPS	CLHS2LP	CLLP2HS	CLFROMULPS	CLTOULPS	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R0W	R0W	R0W	R0W	R0W	R0W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DL0TX2RX	DL0RX2TX	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R0W	R0W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	DLFROMULPS	0	R0W	PLSR.DLFROMULPS Clear Set to “1” to clear PLSR.DLFROMULPS. Writing “0” has no effect.
28	DLTOULPS	0	R0W	PLSR.DLTOULPS Clear Set to “1” to clear PLSR.DLTOULPS. Writing “0” has no effect.
27	CLHS2LP	0	R0W	PLSR.CLHS2LP Clear Set to “1” to clear PLSR.CLHS2LP. Writing “0” has no effect.
26	CLLP2HS	0	R0W	PLSR.CLLP2HS Clear Set to “1” to clear PLSR.CLLP2HS. Writing “0” has no effect.
25	CLFROMULPS	0	R0W	PLSR.CLFROMULPS Clear Set to “1” to clear PLSR.CLFROMULPS. Writing “0” has no effect.
24	CLTOULPS	0	R0W	PLSR.CLTOULPS Clear Set to “1” to clear PLSR.CLTOULPS. Writing “0” has no effect.
23 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	DL0TX2RX	0	R0W	PLSR.DL0TX2RX Clear Set to “1” to clear PLSR.DL0TX2RX. Writing “0” has no effect.
12	DL0RX2TX	0	R0W	PLSR.DL0RX2TX Clear Set to “1” to clear PLSR.DL0RX2TX. Writing “0” has no effect.
11 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.43 Physical Lane Interrupt Enable Register (PLIER)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DLFROMULPS	DLTOULPS	CLHS2LP	CLLP2HS	CLFROMULPS	CLTOULPS	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DL0TX2RX	DL0RX2TX	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	DLFROMULPS	0	RW	Interrupt enable for the all data lane transition from ULPS 0: Disable interrupt of dsi_int_ppi when PLSR.DLFROMULPS =1 1: Enable interrupt of dsi_int_ppi when PLSR.DLFROMULPS =1
28	DLTOULPS	0	RW	Interrupt enable for the all data lane transition to ULPS 0: Disable interrupt of dsi_int_ppi when PLSR.DLTOULPS =1 1: Enable interrupt of dsi_int_ppi when PLSR.DLTOULPS =1
27	CLHS2LP	0	RW	Interrupt enable for the clock lane transition to LP from HS 0: Disable interrupt of dsi_int_ppi when PLSR.CLHS2LP=1 1: Enable interrupt of dsi_int_ppi when PLSR.CLHS2LP=1
26	CLLP2HS	0	RW	Interrupt enable for the clock lane transition to HS from LP 0: Disable interrupt of dsi_int_ppi when PLSR.CLLP2HS=1 1: Enable interrupt of dsi_int_ppi when PLSR.CLLP2HS=1
25	CLFROMULPS	0	RW	Interrupt enable for the clock lane transition from ULPS 0: Disable interrupt of dsi_int_ppi when PLSR.CLFROMULPS=1 1: Enable interrupt of dsi_int_ppi when PLSR.CLFROMULPS=1
24	CLTOULPS	0	RW	Interrupt enable for the clock lane transition to ULPS 0: Disable interrupt of dsi_int_ppi when PLSR.CLTOULPS=1 1: Enable interrupt of dsi_int_ppi when PLSR.CLTOULPS=1
23 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	DL0TX2RX	0	RW	Interrupt enable for the data lane 0 change direction from Tx to Rx 0: Disable interrupt of dsi_int_ppi when PLSR.DL0TX2RX =1 1: Enable interrupt of dsi_int_ppi when PLSR.DL0TX2RX =1
12	DL0RX2TX	0	RW	Interrupt enable for the data lane 0 change direction from Rx to Tx 0: Disable interrupt of dsi_int_ppi when PLSR.DL0RX2TX =1 1: Enable interrupt of dsi_int_ppi when PLSR.DL0RX2TX =1
11 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.44 Video-Input Channel 1 Set 0 Register (VICH1SET0R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VSEN	—	HFPNOLP	HBPNOLP	HSANOLP	—	—	—	—	—	—	VSTPAFT	VSTART
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13	—	0	RW	Reserved When read, the initial value is read. The written value will be ignored.
12	VSEN	0	RW	Video Skew Calibration Enable Skew Calibration is not supported. This bit should be fixed at 0.
11	—	0	RW	Reserved When read, the initial value is read. The written value will be ignored.
10	HFPNOLP	0	RW	HFP period No LP Set suppress not to transition LP during HFP period. 0: Does not suppress transition to LP during the period 1: Suppress transition to LP during the period and keep HS. HS is kept by sending blanking packet.
9	HBPNOLP	0	RW	HBP period No LP Set suppress not to transition LP during HBP period. 0: Does not suppress transition to LP during the period 1: Suppress transition to LP during the period and keep HS. HS is kept by sending blanking packet.
8	HSANOLP	0	RW	HSA period No LP Set suppress not to transition LP during HSA period. When TXESYNC=0, this field has no meaning. 0: Does not suppress transition to LP during the period 1: Suppress transition to LP during the period and keep HS. HS is kept by sending blanking packet.
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	VSTPAFT	0	R0W	Video-Input signals stop after Video-Input Operation stop Stop Video-Input Operation. Video-Input Operation will end at the end of video-frame. After Video-Input Operation end, stop Video-Input signals. If the Video-Input operation is stopped, VICH1SR.RUNNING bit will change to 0. Set “1” to both VSTPAFT and VSTART is prohibited. Set “1” during VICH1SR.RUNNING=0 is prohibited.

Bit	Bit Name	Initial Value	R/W	Description
0	VSTART	0	R0W	Video-Input Operation Start Write 1 to start Video-Input Operation. Set “1” to both VSTART and VSTPAFT is prohibited. Set “1” during VICH1SR.RUNNING=1 is prohibited.

34.3.1.45 Video-Input Channel 1 Set 1 Register (VICH1SET1R)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSPC	—	BPP						CHBUFSZ				—	—	—	—
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DLY												—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CSPC	0	R	Color space of input video pixel This field reflects color space of VICH1PPSETR.DT selected data type. 0: RGB 1: Reserved
30	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 24	BPP	H’10	R	Bit Per Pixel of input video pixel This field reflects bits per pixel of VICH1PPSETR.DT selected data type. H’10: Reserved H’12: 18bpp H’18: 24bpp, 18bpp Loosely (RGB)
23 to 20	CHBUFSZ	H’6	R	Channel Buffer Size H’6: 8kB
19 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17 to 16	—	H’3	R	Reserved When read, the initial value is read. The written value will be ignored.
15 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 2	DLY	All 0	RW	Delay Value The delay Time is calculated by formula of DLY * (period of high-speed serial UI * 32). Setting 0 to this field is prohibited. If VICH1SET0R.HFPNOLP=1, set this field more than HFP period. If VICH1SET0R.HBPNOLP=1, set this field more than HBP period. If VICH1SET0R.HSANOLP=1, set this field more than HSA period.
1 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.46 Video-Input Channel 1 Status Register (VICH1SR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	VBUFOVF	VBUFUDF	—	TIMERR	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	VIRDY	RUNNING	STOP	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.
25, 24	—	00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	VBUFOVF	0	R	Video-Input Buffer Overflow Error Data overflow is occurred in Video-Input Buffer. Write “1” to VICH1SCR.VBUFOVF register to clear this field.
22	VBUFUDF	0	R	Video-Input Buffer Underflow Error Data underflow is occurred in Video-Input Buffer. Write “1” to VICH1SCR.VBUFUDF register to clear this field.
21	—	0	R	Reserved When read, the initial value is read. The written value will be ignored.
20	TIMERR	0	R	Timing Error Timing error is occurred during video input operation. Write “1” to VICH1SCR.TIMERR register to clear this field.
19 to 15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	—	0	R	Reserved When read, the initial value is read. The written value will be ignored.
13 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	VIRDY	0	R	Video-Input Signal Accept Ready The value will be set to 1 when the input signal can accept by Video-Input Operation channel 1 ready.
2	RUNNING	0	R	Video-Input Operation channel 1 Transfer Running The value will be set to 1 during the Video-Input Operation channel 1 Transfer running. VICH1SET0R, VICH1SET1R, VICH1PPSETR, VICH1VSSETR, VICH1VPSETR, VICH1HSSETR and VICH1HPSETR are prohibited to change value during running (VICH1SR.RUNNING=1).
1	STOP	0	R	Video Input Operation channel 1 Transfer Stop Write “1” to VICH1SCR.STOP register to clear this field.
0	START	0	R	Video Input Operation channel 1 Transfer Start. Write “1” to VICH1SCR.START register to clear this field.

34.3.1.47 Video-Input Channel 1 Status Clear Register (VICH1SCR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	VBUFOVF	VBUFUDF	—	TIMERR	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R0W	R0W	R0W	R0W	R	R	R0W	R0W	R0W	R0W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	VIRDY	—	STOP	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R	R	R	R	R	R	R	R	R	R	R0W	R	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 26	—	All 0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	VBUFOVF	0	R0W	VICH1SR.VBUFOVF Clear Set to "1" to clear VICH1SR.VBUFOVF. Writing "0" has no effect.
22	VBUFUDF	0	R0W	VICH1SR.VBUFUDF Clear Set to "1" to clear VICH1SR.VBUFUDF. Writing "0" has no effect.
21	—	0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	TIMERR	0	R0W	VICH1SR.TIMERR Clear Set to "1" to clear VICH1SR.TIMERR. Writing "0" has no effect.
19 to 15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	—	0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	VIRDY	0	R0W	VICH1SR.VIRDY Clear Set to "1" to clear VICH1SR.VIRDY. Writing "0" has no effect.
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	STOP	0	R0W	VICH1SR.STOP Clear Set to "1" to clear VICH1SR.STOP. Writing "0" has no effect.
0	START	0	R0W	VICH1SR.START Clear Set to "1" to clear VICH1SR.START. Writing "0" has no effect.

34.3.1.48 Video-Input Channel 1 Interrupt Enable Register (VICH1IER)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	VBUFOVF	VBUFUDF	—	TIMERR	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	VIRDY	—	STOP	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	R	R	R	R	R	R	R	R	R	R	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 26	—	All 0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23	VBUFOVF	0	RW	Interrupt enable for the Video-Input Buffer Overflow Error 0: Disable interrupt of dsi_int_vin1 when VICH1SR.VBUFOVF=1 1: Enable interrupt of dsi_int_vin1 when VICH1SR.VBUFOVF=1
22	VBUFUDF	0	RW	Interrupt enable for the Video-Input Buffer Underflow Error 0: Disable interrupt of dsi_int_vin1 when VICH1SR.VBUFUDF=1 1: Enable interrupt of dsi_int_vin1 when VICH1SR.VBUFUDF=1
21	—	0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	TIMERR	0	RW	Interrupt enable for the Timing Error 0: Disable interrupt of dsi_int_vin1 when VICH1SR.TIMERR=1 1: Enable interrupt of dsi_int_vin1 when VICH1SR.TIMERR=1
19 to 15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	—	0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	VIRDY	0	RW	Interrupt enable for the Video Input Accept Ready 0: Disable interrupt of dsi_int_vin1 when VICH1SR.VIRDY=1 1: Enable interrupt of dsi_int_vin1 when VICH1SR.VIRDY=1
2	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	STOP	0	RW	Interrupt enable for the Video Input Operation channel 1Transfer Stop 0: Disable interrupt of dsi_int_vin1 when VICH1SR.STOP=1 1: Enable interrupt of dsi_int_vin1 when VICH1SR.STOP=1
0	START	0	RW	Interrupt enable for the Video Input Operation channel 1 Transfer Start 0: Disable interrupt of dsi_int_vin1 when VICH1SR.START=1 1: Enable interrupt of dsi_int_vin1 when VICH1SR.START=1

34.3.1.49 Video-Input Channel 1 Pixel Packet Set Register (VICH1PPSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	VC		DT					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXESYNC	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23, 22	VC	All 0	RW	Video-Input Operation Channel 1 Pixel Stream Packet Header Virtual Channel Identifier
21 to 16	DT	H'0E	RW	Video-Input Operation Channel 1 Pixel Stream Packet Header Data Type Following values are allowed to set. H'0E: Reserved H'1E: Packed Pixel Stream, 18bit RGB H'2E: Loosely Packed Pixel Stream, 18bit RGB H'3E: Packed Pixel Stream, 24bit RGB Other Setting values are prohibited
15	TXESYNC	0	RW	Transmit End of Sync Pulse 0: HSE and VSE are NOT transmitted 1: HSE and VSE are transmitted "0" setting is used for "Burst Mode" sequence or "Non-Burst Mode with Sync Events" sequence. "1" setting is used for "Non-Burst Mode with Sync Pulse" sequence.
14 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	—	All 0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

34.3.1.50 Video-Input Channel 1 Vertical Size Set Register (VICH1VSSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	VACTIVE														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSPOL	—	—	—	VSA											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 16	VACTIVE	All 0	RW	Set Line count of VACTIVE (Vertical Active lines)
15	VSPOL	0	RW	Polarity of VSYNC Set polarity of VSYNC signal. 0: High Active 1: Low Active
14 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	VSA	All 0	RW	VSA Count Set Line count of VSA (Vertical Sync Active)

34.3.1.51 Video-Input Channel 1 Vertical Porch Set Register (VICH1VPSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VFP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VBP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	VFP	All 0	RW	VFP Count Set Line count of VFP (Vertical Front Porch)
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	VBP	All 0	RW	VBP Count Set Line count of VBP (Vertical Back Porch)

34.3.1.52 Video-Input Channel 1 Horizontal Size Set Register (VICH1HSSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	HACTIVE														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HSPOL	—	—	—	HSA											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 16	HACTIVE	All 0	RW	HACTIVE Count Set pixel count of HACTIVE (Horizontal Active pixels)
15	HSPOL	0	RW	Polarity of HSYNC Set polarity of HSYNC signal. 0: High Active 1: Low Active.
14 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	HSA	All 0	RW	HSA Count Set pixel count of HSA (Horizontal Sync Active)

34.3.1.53 Video-Input Channel 1 Horizontal Porch Set Register (VICH1HPSETR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	HFP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HBP												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	HFP	All 0	RW	HFP Count Set pixel count of HFP (Horizontal Front Porch)
15 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	HBP	All 0	RW	HBP Count Set pixel count of HBP (Horizontal Back Porch)

34.3.1.54 Sequence Channel x Set 0 Register (SQCHxSET0R)

“x” indicates channel number.

“x” is 0 or 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	RW	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	R	R	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23	—	1	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
22 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8 to 4	—	All 0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3, 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	START	0	R0W	Sequence Operation Start Write 1 to start running from Descriptor #0. Set “1” during RUNNING=1 is prohibited. Writing “0” has no effect.

34.3.1.55 Sequence Channel x Set 1 Register (SQCHxSET1R)

“x” indicates channel number.

“x” is 0 or 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAXDESNUM								CHBUFSZ				—	—	—	—
Initial Value	0	0	0	0	1	0	0	0	—	—	—	—	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MAXDESNUM	H’08	RW	Max Descriptor Number This field indicate maximum number of descriptors. Writing value except the initial value is prohibited. (e.g. Setting 8 means descriptor can use #0 to #7)
23 to 20	CHBUFSZ	H’0 or H’3	R	Sequence Channel x Buffer Size H’0: 128B (x = 0 case) H’3: 1kB (x = 1 case) Setting transfer packet size larger than this field is prohibited.
19 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.56 Sequence Channel x Status Register (SQCHxSR)

“x” indicates channel number.

“x” is 0 or 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	—	TXIBERR	—	—	—	—	PKTBIGERR	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADESFIN	—	—	—	AACTFIN	—	RUNNING	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	RXAKE	0	R	Receive Acknowledge and Error Report Packet RXSR.RXAKE is also reported. Write “1” to SQCHxSCR.RXAKE register to clear this field.
29	RXCORERR	0	R	Receive Correctable Error Received packet has Correctable Error. RXSR.ECCERR1B is also reported. Write “1” to SQCHxSCR.RXCORERR register to clear this field.
28	RXPKTDFAIL	0	R	Receive Packet Data Fail Packet Header Saved Correctly but Data did not save correctly. One or more of the following bits are also reported. RXSR.CRCERR, RXSR.WCERR, RXSR.UEXPKTERR, RXSR.MAXRPSZERR, RXSR.RXOVFERR or RXSR.IBERR. Write “1” to SQCHxSCR.RXPKTDFAIL register to clear this field.
27	RXFAIL	0	R	Receive Fail Expected receive did not done. One or more of the following bits are also reported. RXSR.PRESPTOERR, RXSR.ECCERR, RXSR.MLFERR or RXSR.NORETERR. Write “1” to SQCHxSCR.RXFAIL register to clear this field.
26	RXFATALERR	0	R	Receive Fatal Error Fatal timeout occurred during BTA. One or more of the following bits are also reported. FERRSR.TATO or FERRSR.LRXHTO. Write “1” to SQCHxSCR.RXFATALERR register to clear this field.
25	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	TXIBERR	0	R	Tx Internal Bus Error When this value is “1”, the Internal Bus Read had failed response. Write “1” to SQCHxSCR.TXIBERR register to clear this field.
23	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	—	0	R	Reserved When read, the initial value is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
21, 20	—	00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	PKTBIGERR	0	R	Packet is too big Error Sequence operation channel 0: Hard wired to 0. Sequence operation channel 1: This bit is asserted when length of sequence packet is bigger than Video-Input channel’s BLLP. Write “1” to SQCHxSCR.PKTBIGERR register to clear this field.
18, 17	—	00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16, 15	—	00	R	Reserved When read, the initial value is read. The written value will be ignored.
14 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	ADESFIN	0	R	All Descriptor Finished by Setting Descriptor finished by descriptor SQCHxDSCyAR.NXACT=H’0 setting. Write “1” to SQCHxSCR.ADESFIN register to clear this field.
7 to 5	—	000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	AACTFIN	0	R	Descriptor’s All Action Finished This bit is set if this descriptor’s all action finished with SQCHxDSCyCR.FINACT[0] =1b. Write “1” to SQCHxSCR.AACTFIN register to clear this field.
3	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RUNNING	0	R	Sequence Operation Running The value will be set to 1 during the sequence operation running. Do not change descriptor values while this bit is 1. Do not change the values in SQCHxSET0R and SQCHxSET1R.
1, 0	—	00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.57 Sequence Channel x Status Clear Register (SQCHxSCR)

“x” indicates channel number.

“x” is 0 or 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	—	TXIBERR	—	—	—	—	PKTBIGERR	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R0W	R0W	R0W	R0W	R	R0W	R	R0W	R	R	R0W	R	R	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADESFIN	—	—	—	AACTFIN	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W	R	R	R	R	R	R	R0W	R	R	R	R0W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	RXAKE	0	R0W	SQCHxSR.RXAKE Clear Set to “1” to clear SQCHxSR.RXAKE. Writing “0” has no effect.
29	RXCORERR	0	R0W	SQCHxSR.RXCORERR Clear Set to “1” to clear SQCHxSR.RXCORERR. Writing “0” has no effect.
28	RXPKTDFAIL	0	R0W	SQCHxSR.RXPKTDFAIL Clear Set to “1” to clear SQCHxSR.RXPKTDFAIL. Writing “0” has no effect.
27	RXFAIL	0	R0W	SQCHxSR.RXFAIL Clear Set to “1” to clear SQCHxSR.RXFAIL. Writing “0” has no effect.
26	RXFATALERR	0	R0W	SQCHxSR.RXFATALERR Clear Set to “1” to clear SQCHxSR.RXFATALERR. Writing “0” has no effect.
25	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	TXIBERR	0	R0W	SQCHxSR.TXIBERR Clear Set to “1” to clear SQCHxSR.TXIBERR. Writing “0” has no effect.
23	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	—	0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21, 20	—	00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	PKTBIGERR	0	R0W	SQCHxSR.PKTBIGERR Clear Set to “1” to clear SQCHxSR.PKTBIGERR. Writing “0” has no effect.
18, 17	—	00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
16, 15	—	00	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
14 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	ADESFIN	0	R0W	SQCHxSR.ADESFIN Clear Set to “1” to clear SQCHxSR.ADESFIN. Writing “0” has no effect.
7 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	AACTFIN	0	R0W	SQCHxSR.AACTFIN Clear Set to “1” to clear SQCHxSR.AACTFIN. Writing “0” has no effect.
3 to 0	START	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.58 Sequence Channel x Interrupt Enable Register (SQCHxIER)

“x” indicates channel number.

“x” is 0 or 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	—	TXIBERR	—	—	—	—	PKTBIGERR	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	R	RW	R	RW	R	R	RW	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADESFIN	—	—	—	AACTFIN	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	RW	R	R	R	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30	RXAKE	0	RW	SInterrupt enable for the Receive Acknowledge and Error Report Packet 0: Disable interrupt of dsi_int_sqx when SQCHxSR.RXAKE=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.RXAKE=1
29	RXCORERR	0	RW	Interrupt enable for the Receive Correctable Error 0: Disable interrupt of dsi_int_sqx when SQCHxSR.RXCORERR=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.RXCORERR=1
28	RXPKTDFAIL	0	RW	Interrupt enable for the Receive Packet Data Fail 0: Disable interrupt of dsi_int_sqx when SQCHxSR.RXPKTDFAIL=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.RXPKTDFAIL=1
27	RXFAIL	0	RW	Interrupt enable for the Receive Fail 0: Disable interrupt of dsi_int_sqx when SQCHxSR.RXFAIL=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.RXFAIL=1
26	RXFATALERR	0	RW	Interrupt enable for the Receive Fatal Error 0: Disable interrupt of dsi_int_sqx when SQCHxSR.RXFATALERR=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.RXFATALERR=1
25	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	TXIBERR	0	RW	Interrupt enable for the Tx Internal Bus Error 0: Disable interrupt of dsi_int_sqx when SQCHxSR.TXIBERR=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.TXIBERR=1
23	—	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22	—	0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21, 20	—	00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	PKTBIGERR	0	RW	Interrupt enable for the Packet is too big Error Sequence operation channel 0: This field has no meaning. Sequence operation channel 1: 0: Disable interrupt of dsi_int_sqx when SQCHxSR.PKTBIGERR=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.PKTBIGERR=1

Bit	Bit Name	Initial Value	R/W	Description
18, 17	—	00	RW	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16, 15	—	00	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
14 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	ADESFIN	0	RW	Interrupt enable for the Descriptor Finished by Setting 0: Disable interrupt of dsi_int_sqx when SQCHxSR.ADESFIN=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.ADESFIN=1
7 to 5	—	000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	AACTFIN	0	RW	Interrupt enable for the Descriptor’s All Action Finished 0: Disable interrupt of dsi_int_sqx when SQCHxSR.AACTFIN=1 1: Enable interrupt of dsi_int_sqx when SQCHxSR.AACTFIN=1
3 to 0	START	0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

34.3.1.59 Sequence Channel x Descriptor y-A Register (SQCHxDSCyAR)

"x" indicates channel number.

"x" is 0 or 1.

"y" indicates descriptor number.

"y" is 0 to 7.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	NXACT		BTA		SPD	FMT	VC		DT					
Initial Value	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA1								DATA0							
Initial Value	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2	*2
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	*2	RW	Reserved When read, the last written value is read. When writing, be sure to write 0. Operation is not guaranteed if a value other than 0 is written.
29 to 28	NXACT	*2	RW	Next Action H'0: Terminate Operation after this descriptor finished. H'1: Operate next descriptor after this descriptor finished H'2: Reserved H'3: Reserved
27 to 26	BTA	*2	RW	Bus Turn Around Set Tx Request or no-operation with Bus Turn-Around (BTA) or not. H'0: Tx Request without BTA or no-operation. H'1: Tx non-Read Request with BTA. H'2: Tx Read Request with BTA. H'3: BTA only
25	SPD	*2	RW	Speed Set the speed type in Tx Request. 0: High Speed 1: Low Power Prohibited to set this bit 1 during Video-Input Operation running.
24	FMT	*2	RW	Format Packet format of Tx Packet Header. This field should be set by the data type. 0: Short Packet 1: Long Packet
23 to 22	VC	*2	RW	Virtual Channel Identifier of virtual channel of Tx Packet Header.
21 to 16	DT	*2	RW	Data Type Data type of Tx Packet Header.
15 to 8	DATA1	*2	RW	Data 1 Data1 of Tx Packet Header.*1 When Tx Packet format is Long Packet, this value is upper 8 bits of the word count.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	DATA0	*[2]	RW	Data 0 Data0 of Tx Packet Header.*[1] When Tx Packet format is Long Packet, this value is lower 8 bits of the word count.

Note 1. • The maximum size for Command Transfer mode Long Packet by using Register is 16 Byte.
(SQCHxDSCyBR.DTSEL = H’0 and SQCHxDSCyAR.FMT=1)
• The maximum size for Command Transfer mode Long Packet by using internal bus memory is SQCHxSET1R.CHBUFSZ Byte.
(SQCHxDSCyBR.DTSEL = H’1 and SQCHxDSCyAR.FMT=1)

Note 2. Initial Values are determined by initial value of Descriptor RAM.

34.3.1.60 Sequence Channel x Descriptor y-B Register (SQCHxDSCyBR)

“x” indicates channel number.

“x” is 0 or 1.

“y” indicates descriptor number.

“y” is 0 to 7.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DTSEL		—	—	—	—	—	—	—	—
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	*1	RW	Reserved When read, the last written value is read. When writing, be sure to write 0. Operation is not guaranteed if a value other than 0 is written.
25 to 24	DTSEL	*1	RW	Data Select H’0: Packet Payload Data register H’1: Long Packet data use memory space. In the case of write. Internal Read will be cause to SQCHxDSCyDR address. In the case of Read. Internal Write will be cause to SQCHxDSCyDR address. *Note*: Read Header always saved to RXRSSxR register. H’2: Reserved H’3: Reserved If write request with short Packet, then set H’0. When DTSEL=H’0, Long Packet write request data size shall be same or lower than 16Byte.When read request data to save has over 16 Byte word count, it reports as error.
23 to 0	—	*1	RW	Reserved When read, the last written value is read. When writing, be sure to write 0. Operation is not guaranteed if a value other than 0 is written.

Note 1. Initial Values are determined by initial value of Descriptor RAM.

34.3.1.61 Sequence Channel x Descriptor y-C Register (SQCHxDSCyCR)

“x” indicates channel number.

“x” is 0 or 1.

“y” indicates descriptor number.

“y” is 0 to 7.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ACTCODE								—	AUXOP	—	—	—	—	—	—
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	FINACT		
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ACTCODE	*1	RW	Action Code When BTA action, indicate rx result save slot number. Setting value equal or more than 4 is prohibited. Setting same rx result save slot number value to different descriptor is prohibited. When AUXOP is 1. H’00: Send Reset-Trigger. H’01: Reserved H’02: Reserved H’03: Reserved H’08: No-operation Other: Set H’00 in other case.
23	—	*1	RW	Reserved When read, the last written value is read. When writing, be sure to write 0. Operation is not guaranteed if a value other than 0 is written.
22	AUXOP	*1	RW	Auxiliary Operation Use for auxiliary operation execution. When this bit is set to 1, SQCHxDSCyBR.BTA should be set to H’0. Set this bit 1 with BTA=H’0. Prohibited to set this bit 1 during Video-Input Operation running.
21 to 3	—	*1	RW	Reserved When read, the last written value is read. When writing, be sure to write 0. Operation is not guaranteed if a value other than 0 is written.
2 to 0	FINACT	*1	RW	Finish Action bit 0: set SQCHxSR.AACTFIN if this descriptor’s all action finished. (0b: Disable, 1b: Enable). Since each descriptor’s operation is not same, descriptors may finish out-of-order. For example, Descriptor#1 may be finished before Descriptor#0.

Note 1. Initial Values are determined by initial value of Descriptor RAM.

34.3.1.62 Sequence Channel x Descriptor y-D Register (SQCHxDSCyDR)

“x” indicates channel number.

“x” is 0 or 1.

“y” indicates descriptor number.

“y” is 0 to 7.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LADDR															
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LADDR															
Initial Value	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1	*1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LADDR	*1	RW	Lower Address Set Lower 32 bits address for long packet Payload Data in Sequence Operation channel x Descriptor y. When Sequence Operation Data Select is “use memory space” (SQCHxDSCyBR.DTSEL =H’1) and Sequence Operation Format is Long packet (SQCHxDSCyAR.FMT=1b), the DSI-Tx Module accesses the address by internal bus Master Read in Write action and internal bus Write in Read action. Set lower three bits to “000b”.

Note 1. Initial Values are determined by initial value of Descriptor RAM.

34.3.2 Resisters in D-PHY

34.3.2.1 DSI D-PHY Control Register 0 (DSIDPHYCTRL0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CAL_EN_HSRX_OFS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CMN_MASTER_EN	—	—	—	—	—	RE_VDD_DETVCCQLV18	EN_LDO1200	EN_BGR
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CAL_EN_HSRX_OFS	1	RW	HS-RX Offset Calibration enable 0b: HS-RX Start with Offset Calibration. 1b: HS-RX Start without Offset Calibration.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	CMN_MASTER_EN	1	RW	Select D-PHY mode master or slave of All-lane. 1b: Master side DSI-v1.3.1 (D-PHY v2.1 CIL_MCNN,CIL_MFAA) 0b: Slave side CSI2-v1.3 (D-PHY v2.0 CIL_SCNN,CIL_SFEN)
7 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	RE_VDD_DETVCCQLV18	1	RW	When VCCQLV18 is OFF, fix the terminal on the interface of the different power supply to Low. 1b: VCCQLV18 ON 0b: VCCQLV18 OFF
1	EN_LDO1200	0	RW	LDO1200 Enable 1b: Enable 0b: Disable
0	EN_BGR	0	RW	BGR Enable 1b: Enable 0b: Disable

34.3.2.2 DSI D-PHY Timing Register 0 (DSIDPHYTIM0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCLK_MISS								—	—	—	—	—	T_INIT		
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	T_INIT															
Initial Value	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TCLK_MISS	1	RW	Timing parameter Set 0.
23 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18 to 0	T_INIT	10000	RW	Timing parameter*[1]

Note 1. **Table 34.6** lists the recommended setting values.

Table 34.6 Recommended Setting Values of the DSI D-PHY Timing Register

No.	Pin Name	Transmission Rate						Note
		80 Mbps	125 Mbps	250 Mbps	360 Mbps	720 Mbps	Over 720 Mbps	
1	T_INIT[18:0]	19’d79801 (H’137B9)	19’d79801 (H’137B9)	19’d79801 (H’137B9)	19’d79801 (H’137B9)	19’d79801 (H’137B9)	19’d79801 (H’137B9)	Min: 100 µs Recommend: 600 µs
2	TCLK_PREPARE[7:0]	8’d8 (H’8)	8’d8 (H’8)	8’d8 (H’8)	8’d8 (H’8)	8’d8 (H’8)	8’d8 (H’8)	
3	THS_PREPARE[7:0]	8’d13 (H’D)	8’d12 (H’C)	8’d12 (H’C)	8’d10 (H’A)	8’d9 (H’9)	8’d9 (H’9)	
4	TCLK_ZERO[7:0]	8’d33 (H’21)	8’d33 (H’21)	8’d33 (H’21)	8’d33 (H’21)	8’d33 (H’21)	8’d33 (H’21)	
5	TCLK_PRE[7:0]	8’d24 (H’18)	8’d15 (H’F)	8’d13 (H’D)	8’d4 (H’4)	8’d4 (H’4)	8’d4 (H’4)	
6	TCLK_POST[7:0]	8’d94 (H’5E)	8’d94 (H’5E)	8’d94 (H’5E)	8’d35 (H’23)	8’d35 (H’23)	8’d35 (H’23)	
7	TCLK_TRAIL[7:0]	8’d10 (H’A)	8’d10 (H’A)	8’d10 (H’A)	8’d7 (H’7)	8’d7 (H’7)	8’d7 (H’7)	
8	THS_ZERO[7:0]	8’d23 (H’17)	8’d23 (H’17)	8’d23 (H’17)	8’d16 (H’10)	8’d16 (H’10)	8’d16 (H’10)	
9	THS_TRAIL[7:0]	8’d17 (H’11)	8’d17 (H’11)	8’d16 (H’10)	8’d9 (H’9)	8’d9 (H’9)	8’d9 (H’9)	
10	THS_EXIT[7:0]	8’d13 (H’D)	8’d13 (H’D)	8’d13 (H’D)	8’d13 (H’D)	8’d13 (H’D)	8’d13 (H’D)	
11	TLPX[7:0]	8’d6 (H’6)	8’d6 (H’6)	8’d6 (H’6)	8’d6 (H’6)	8’d6 (H’6)	8’d6 (H’6)	

Note: The above table lists the recommended setting values. Please calculate and use linearly interpolated value for the setting of non-described in the table.

34.3.2.3 DSI D-PHY Timing Register 1 (DSIDPHYTIM1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	THS_PREPARE								TCLK_PREPARE							
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	THS_SETTLE								TCLK_SETTLE							
Initial Value	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	THS_PREPARE	4	RW	Timing parameter*[1]
23 to 16	TCLK_PREPARE	3	RW	Timing parameter*[1]
16 to 8	THS_SETTLE	9	RW	Timing parameter Set 0.
7 to 0	TCLK_SETTLE	9	RW	Timing parameter Set 0.

Note 1. **Table 34.6** lists the recommended setting values.

34.3.2.4 DSI D-PHY Timing Register 2 (DSIDPHYTIM2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCLK_TRAIL								TCLK_POST							
Initial Value	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCLK_PRE								TCLK_ZERO							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TCLK_TRAIL	5	RW	Timing parameter*[1]
23 to 16	TCLK_POST	16	RW	Timing parameter*[1]
15 to 8	TCLK_PRE	1	RW	Timing parameter*[1]
7 to 0	TCLK_ZERO	25	RW	Timing parameter*[1]

Note 1. **Table 34.6** lists the recommended setting values.

34.3.2.5 DSI D-PHY Timing Register 3 (DSIDPHYTIM3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TLPX								THS_EXIT							
Initial Value	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	THS_TRAIL								THS_ZERO							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TLPX	4	RW	Timing parameter*1
23 to 16	THS_EXIT	9	RW	Timing parameter*1
15 to 8	THS_TRAIL	6	RW	Timing parameter*1
7 to 0	THS_ZERO	11	RW	Timing parameter*1

Note 1. **Table 34.6** lists the recommended setting values.

34.4 Operation

34.4.1 Interrupts

Interrupts can be divided into the following categories:

Table 34.7 Interrupt Signals (1/2)

Interrupt Signal	Category	Interrupt Condition	Interrupt Source Register
dsi_int_sq0	Sequence operation channel 0 interrupt	Receive Acknowledge and Error Report Packet in Sequence Operation BTA	SQCH0SR.RXAKE
		Receive Correctable Error in Sequence Operation BTA	SQCH0SR.RXCORERR
		Receive Packet Data Error in Sequence Operation BTA	SQCH0SR.RXPKTDFAIL
		Receive Fail Error in Sequence Operation BTA	SQCH0SR.RXFAIL
		Receive Fatal Error in Sequence Operation BTA	SQCH0SR.RXFATALERR
		Transmit Internal Bus Error in Sequence Operation	SQCH0SR.TXIBERR
		Descriptor Finished by Setting in Sequence Operation	SQCH0SR.ADESFIN
		Descriptor’s All Action Finished in Sequence Operation	SQCH0SR.AACTFIN
dsi_int_sq1	Sequence operation channel 1 interrupt	Receive Acknowledge and Error Report Packet in Sequence Operation BTA	SQCH1SR.RXAKE
		Receive Correctable Error in Sequence Operation BTA	SQCH1SR.RXCORERR
		Receive Packet Data Error in Sequence Operation BTA	SQCH1SR.RXPKTDFAIL
		Receive Fail Error in Sequence Operation BTA	SQCH1SR.RXFAIL
		Receive Fatal Error in Sequence Operation BTA	SQCH1SR.RXFATALERR
		Transmit Internal Bus Error in Sequence Operation	SQCH1SR.TXIBERR
		Transmit Packet is too big Error in Sequence Operation	SQCH1SR.PKTBIGERR
		Descriptor Finished by Setting in Sequence Operation	SQCH1SR.ADESFIN
		Descriptor’s All Action Finished in Sequence Operation	SQCH1SR.AACTFIN
dsi_int_vin1	Video-Input operation channel 1 interrupt	Transmit Buffer Overflow in Video-Input Operation	VICH1SR.VBUFOVF
		Transmit Buffer Underflow in Video-Input Operation	VICH1SR.VBUFUDF
		Transmit Timing Error in Video-Input Operation	VICH1SR.TIMERR
		Video-Input Signal Accept Ready	VICH1SR.VIRDY
		Video-Input Operation Stopped	VICH1SR.STOP
		Video-Input Operation Started	VICH1SR.START

Table 34.7 Interrupt Signals (2/2)

Interrupt Signal	Category	Interrupt Condition	Interrupt Source Register
dsi_int_rcv	DSI Packet Receive interrupt	Receive Acknowledge and Error Report Packet	RXSR.RXAKE
		ECC 1bit Error	RXSR.ECCERR1B
		Maximum Return Packet Size Error	RXSR.MAXRPSZERR
		No Return Error	RXSR.NORETERR
		Peripheral Response timeout	RXSR.PRESPTOERR
		Receive Buffer overflow error	RXSR.RXOVFERR
		Internal Bus Error detect on data save	RXSR.IBERR
		CRC Error	RXSR.CRCERR
		Word Count Error	RXSR.WCERR
		Unexpected Packet Error	RXSR.UEXPKTERR
		ECC Error	RXSR.ECCERR
		Malformed Error	RXSR.MLFERR
		External Tearing Effect	RXSR.EXTTEDET
		Receive ACK trigger	RXSR.RXACK
		Receive Tearing Effect trigger	RXSR.RXTE
		Receive Reset trigger	RXSR.RXRTRG
		Receive Unknown-5 trigger	RXSR.RXUK5TRG
		Receive EoTp	RXSR.RXEOTP
		Receive Response packet	RXSR.RXRESP
		Turnaround Acknowledge Timeout	RXSR.TATO
		LP-RX Host Processor Timeout	RXSR.LRXHTO
		BTA request End	RXSR.BTAREQEND
dsi_int_ferr	DSI Fatal Error interrupt	Contention Error LP1	FERRSR.ERRCLP1
		Contention Error LP0	FERRSR.ERRCLP0
		Control Error	FERRSR.ERRCTRL
		Sync Escape Error	FERRSR.ERRSYNESC
		Error Escape	FERRSR.ERRESC
		Turnaround Acknowledge Timeout	FERRSR.TATO
		LP-RX Host Processor Timeout	FERRSR.LRXHTO
		HS TX Timeout	FERRSR.HTXTO
dsi_int_ppi	DSI D-PHY PPI interrupt	Data Lanes return from ULPS	PLSR.DLFROMULPS
		Data Lanes go to ULPS	PLSR.DLTOULPS
		Clock Lane change HS to LP	PLSR.CLHS2LP
		Clock Lane change LP to HS	PLSR.CLLP2HS
		Clock Lanes return from ULPS	PLSR.CLFROMULPS
		Clock Lanes go to ULPS	PLSR.CLTOULPS
		Data Lane 0 change TX to RX	PLSR.DL0TX2RX
		Data Lane 0 change RX to TX	PLSR.DL0RX2TX

The interrupt signal consists of combination logic. The interrupt signal from a Status Register is enabled by the signal related Interrupt Enable Register. These interrupt signals are bound by OR logic.

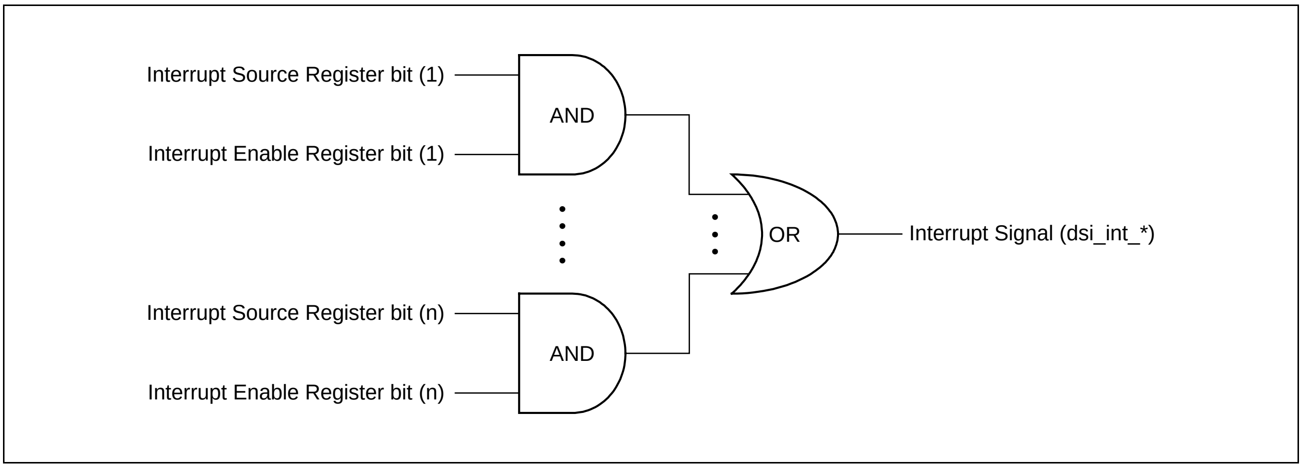


Figure 34.4 Interrupt signal output

34.4.2 Operation Flow

This chapter shows various operation procedures for using the DSI-Tx Module. As for the interrupt operation, Interrupt Enable bit of each interrupt source is enabled (set to 1) before use, and after confirming the interrupt, the interrupt source is cleared. And more, previous action's interrupt source must be cleared before setting the interrupt enable. Flows in this chapter indicate only interrupt bit. The DSI-Tx Module does not care whether firmware use the bits by interrupt action or flag polling action. These descriptions are omitted in each flow in this chapter unless otherwise specified. Error checks are also omitted in normal flow.

34.4.2.1 Reset

(1) Power on Reset and Initial Settings for All Operations

The power on sequence of the DSI-Tx Module is shown **Figure 34.5**.

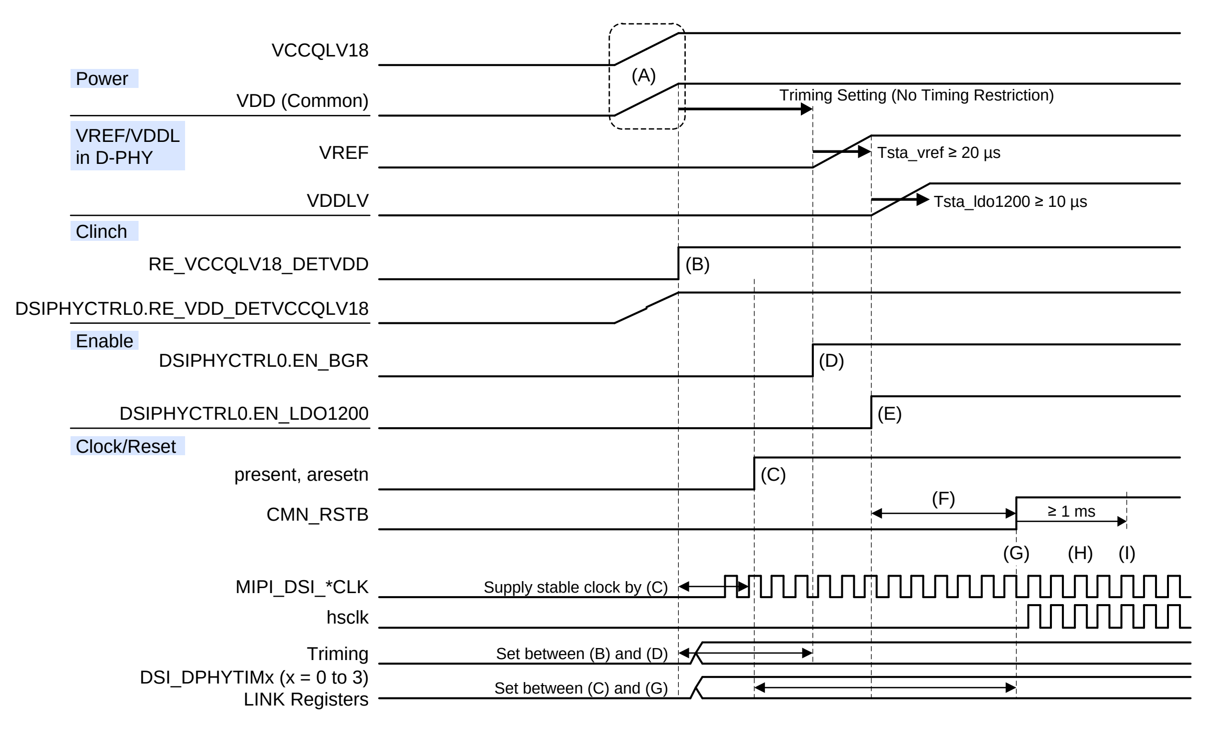


Figure 34.5 Power on sequence

(A) Bring up VDD and VCCQLV18. There are no restrictions on the order to bring up VDD and VCCQLV18.

(B) Set 1 to RE_VCCQLV18_DETVDD.

(C) De-assert preestn and aresetn after after MIPI_DSI_*CLK is stable.

(D) Write 1 to DSIDPHYCTRL0.EN_BGR.

(E) Write 1 to DSIDPHYCTRL0.EN_LDO1200 after waiting for more than 20 usec.

(F) Write DSIDPHYTIMx (x=0 to 3) registers and LINK registers after waiting for more than 10 usec.

(G) De-assert CMN_RSTB.

(H) Wait for more than 1 msec.

(I) The DSI-Tx Module is ready.

At power-on reset, all registers of the DSI-Tx Module are initialized.

Perform the initial settings of the register after canceling the power-on reset. It is not necessary to set the register that is used with the default value.

The registers that need to be initialized are as follows.

- Common Settings
  - TXSETR
  - ULPSSETR
  - DSISETR
  - CLSTPTSETR
  - LPTRNSTSETR
- Timeout Settings
  - PRESPTOBTASETR
  - PRESPTOLPSETR
  - PRESPTOHSSETR
  - HSTXTOSETR
  - LRXHTOSETR
  - TATOSETR

These registers can only be changed during the initial setup after a power-on reset and during a software reset in **Section 34.4.2.1(2)**. It is prohibited to change at any other time

(2) Software Reset

Figure 34.6 shows the operation during software reset. Make sure that APB clock (MIPI_DSI_PCLK), Internal Bus clock (MIPI_DSI_ACLK), Video clock (MIPI_DSI_VCLK), PLL Multiplied clock (MIPI_DSI_PLLCLK), Escape mode Transmit clock (MIPI_DSI_LPCLK) are running and CMN_RSTB is High before starting software reset.

Writing 1 to RSTCR.SWRST will issue a software reset. After issuing the software reset, check the start of the software reset process with RSTSR.SWRSTxx = 1. Then write 1 to RSTCR.FCETXSTP to transition the valid data lane of the D-PHY to the Tx Stop State. By checking RSTSR.DL0DIR = 0 and RSTSR.DLSTPST [x] = 1 of the valid data lane, it is possible to confirm that the valid data lane of D-PHY has transitioned to Tx Stop State. After confirming that the valid data lane of the D-PHY has transitioned to the Tx Stop State, write 0 to RSTCR.FCETXSTP. Then write 0 to RSTCR.SWRST and check the completion of the software reset process with RSTSR.SWRSTx = 0. Since all registers are synchronously reset except for some registers*[1], the software reset process is not completed even if a software reset is issued while the clock is stopped.

Note 1. Registers that require initial settings as described in **Section 34.4.2.1(1)**, GPO0R register, GPO1R register, RSTCR register, RSTSR register, and SQCHxDSCyAR – SQCHxDSCyDR register are not initialized by software reset.

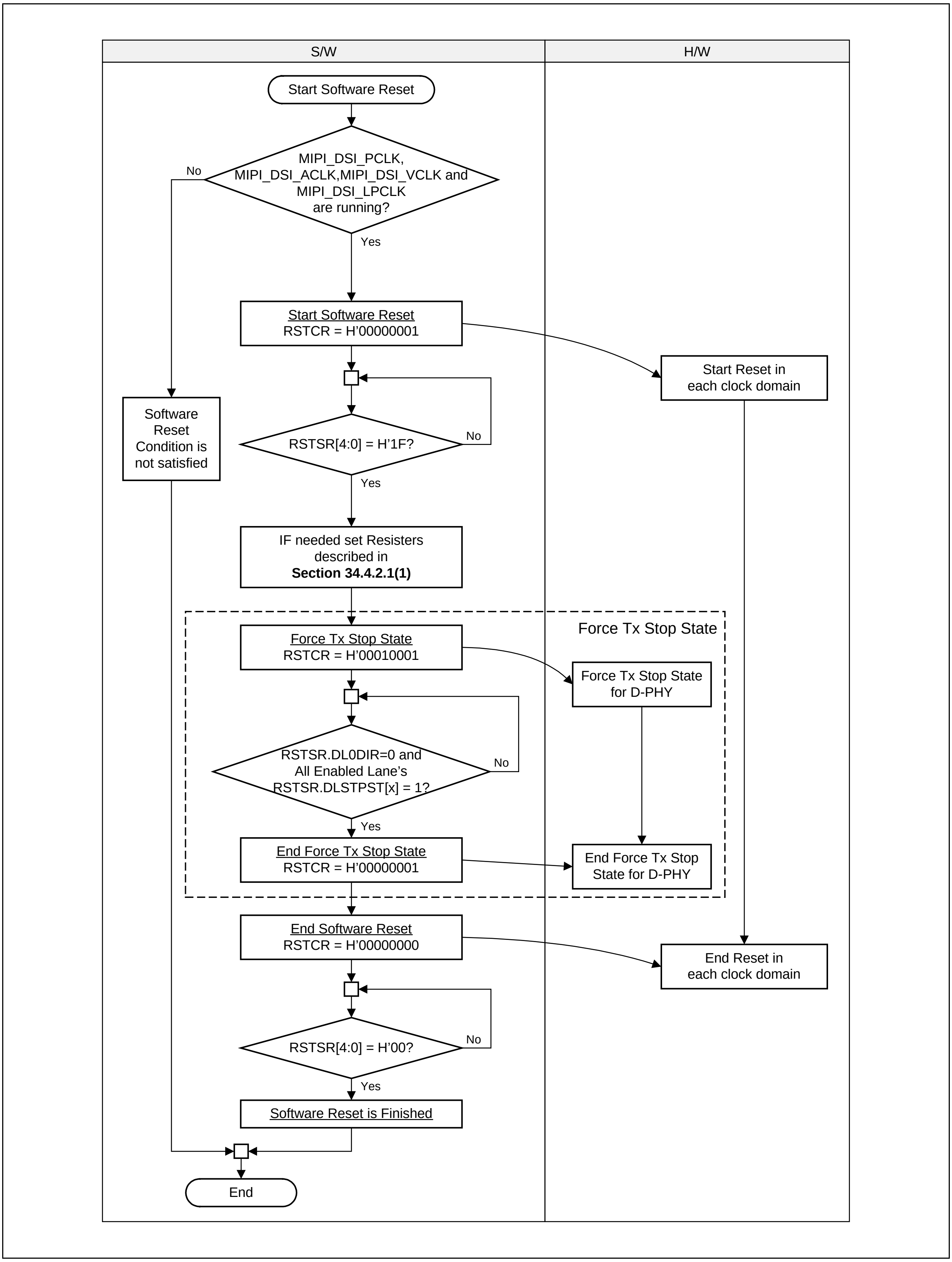


Figure 34.6 Software Reset

34.4.2.2 Start/Stop of HS clock

The following describes the HS Clock Start / Stop sequence. HS Clock Start / Stop is controlled by Software.

Figure 34.7 shows the operation at HS Clock Start.

Write 1 to TXSETR.CLEN to enable DPHY clock lanes before starting Video-Input Operation or Sequence Operation channel 1 operation.

After the clock lane is enabled and the source clock of the clock lane is stable, write 1 to HSCLKSETR.HSCLKRUN to set the clock lane to HS. It is forbidden to change HSCLKRUN while Sequence Channel is in operation.

At the same time as writing 1 to HSCLKSETR.HSCLKRUN, set continuous clock mode and non-continuous clock mode with HSCLKSETR.HSCLKMODE.

In continuous clock mode, always set the clock lane to HS. PLSR.CLLP2HS is set to 1 when the clock lane transitions from LP to HS. After checking, perform the following operations.

In non-continuous clock mode, the DSI-Tx Module automatically transitions the clock lane from LP to HS depending on the presence or absence of HS Transmission. PLSR.CLLP2HS and PLSR.CLHS2LP are set for each transition, but confirmation is not required.

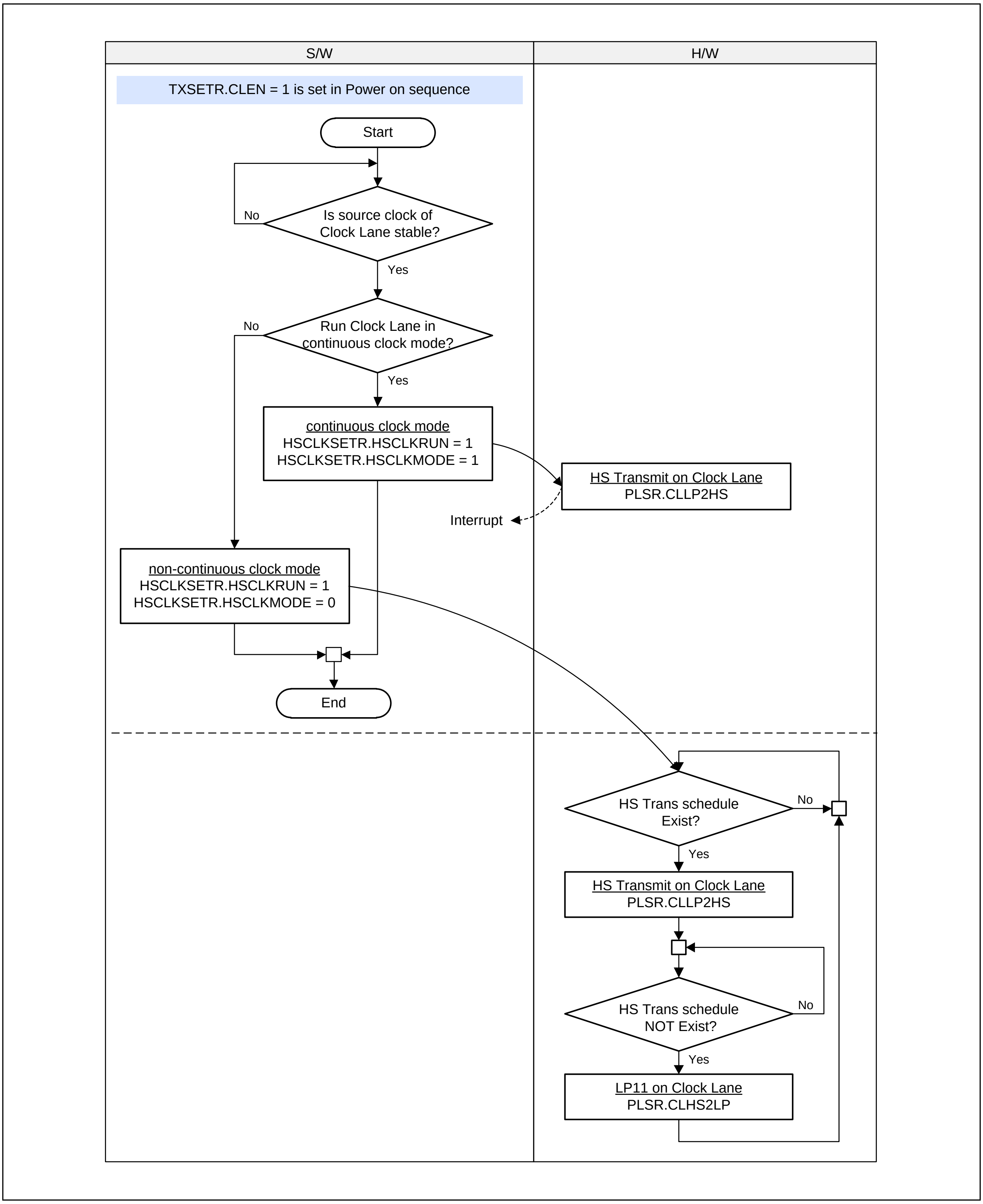


Figure 34.7 Start of HS clock

Figure 34.8 shows the operation when HS Clock Stop. After confirming that Video-Input Operation and Sequence Operation channel 1 are not running, write 0 to HSCLKSETR.HSCLKRUN to make the clock lane LP. PLSR.CLHS2LP is set to 1 when the clock lane transitions from HS to LP. In non-continuous clock mode, you do not need to check PLSR.CLHS2LP because the clock lane is likely to be LP already when you write 0 to HSCLKSETR.HSCLKRUN. It is possible to check the clock lane status by reading PLSR.CLSTPST.

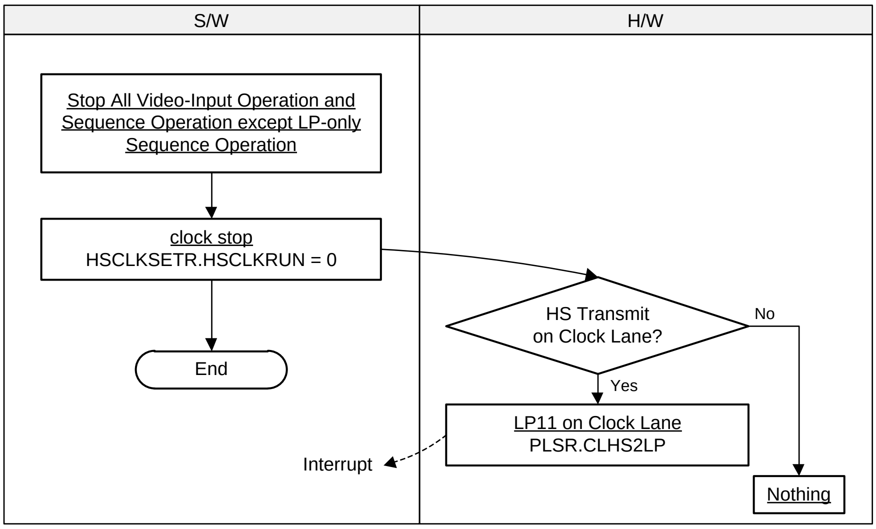


Figure 34.8 Stop of HS clock

34.4.2.3 Sequence Operation

(1) Basic Running Sequence

Sequence Operation requires initial settings before the operation starts.

- Channel Register Setting
  - Set SQCHxSET1R, SQCHxIER
- Prepare Write Data
  - Prepare Write Data to Memory Area
  - Prepare Write Data to TXPPDxR Registers
- Allocate Read Data Area
  - Clear RXRSSR Register if previous valid flag remains.
  - Allocate each read action's Memory area.
- Write all Descriptors
- Confirm running HSCLK if Sequence channel 1 is targeted to run.

When "1" is written to SQCHxSET0R.START, Sequence Operation channel x starts the processing of Descriptor, and when the processing of Descriptor of SQCHxDSCyAR.NXACT = "00b" is completed, the processing of Descriptor ends.

Since Descriptor consists of SRAM, the initial value is indefinite. When the user sets the Descriptor, set all the bits.

After writing 1 to SQCHxSET0R.START on Sequence Operation channel x and SQCHxSR.RUNNING becomes 1, It is prohibited to change the settings of SQCHxSET1R, SQCHxDSCyAR, SQCHxDSCyBR, SQCHxDSCyCR, and SQCHxDSCyDR until SQCHxSR.RUNNING becomes 0.

HS-related operations are prohibited on Sequence Operation channel 0.

(2) Single Packet Transmission

The following describes sending non-read packets using the Sequence Operation channel.

For non-read packet transmission, set the following registers before packet transmission.

SQCHxDSCyAR.NXACT: "00b"

SQCHxDSCyAR.FMT: "0b" Short Packet or "1b" Long Packet

SQCHxDSCyAR.SPD: "0b" High Speed or "1b" Low Speed

SQCHxDSCyAR.BTA: "00b" without BTA or "01b" with BTA

SQCHxDSCyAR.VC, DT, DATA0, DATA1

SQCHxDSCyBR.DTSEL: Specify payload data storage location for non-read packets

"00b" TXPPD0R – TXPPD3R. Payloads size is limited up to 16 Bytes.

"01b" Memory space. Payloads size is limited up to SQCHxSET1R.CHBUFSZ Bytes.

SQCHxDSCyCR.AUXOP "0b"

When setting a Long Packet with SQCHxDSCyAR.FMT, prepare the payload data in the location specified by SQCHxDSCyBR.DTSEL in the form of little endian.

No payload data is required when configuring Short Packets with SQCHxDSCyAR.FMT.

After the above settings, write “1” to SQCHxSET0R.START to send non-read packets.

The user can know the completion of Single Packet Transmission by asserting SQCHxSR.ADESFIN.

If “01b” is set in SQCHxDSCyAR.BTA, BTA is performed after sending a non-read packet.

An ACK trigger or Acknowledge and Error Report Packet may be received when the bus rights are transferred to the peripheral.

The user can get an overview of the reception result by reading the RXRSSxR register with the number specified by SQCHxDSCyCR.ACTCODE. Also, by reading the RXSR register, the user can know the details of the reception result.

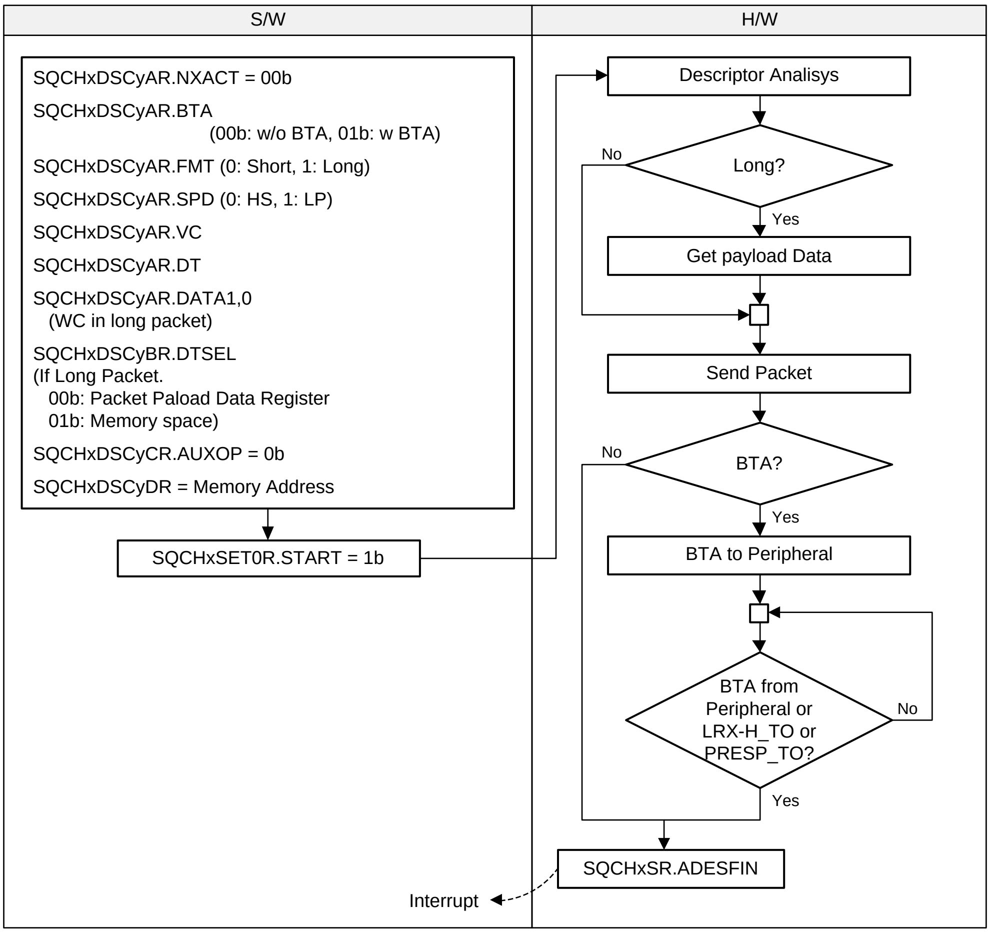


Figure 34.9 Single Packet Transfer

(3) Single Packet Reception

This section describes sending read packets and receiving response packets using the Sequence Operation channel.

Since the read packet is a short packet, set the following registers before sending the packet.

SQCHxDSCyAR.NXACT: “00b”

SQCHxDSCyAR.FMT: “0b” Short Packet

SQCHxDSCyAR.SPD: “0b” High Speed or “1b” Low Speed

SQCHxDSCyAR.BTA: “10b” Read Request with BTA

SQCHxDSCyAR.VC, DT, DATA0, DATA1

SQCHxDSCyBR.DTSEL: Specify the payload data storage location of the response packet

“00b” RXPPD0R – RXPPD3R. Payloads size is limited up to 16 Bytes.

“01b” Memory space. Payloads size is limited up to RXBUFSZR.RXBUFSZ Bytes.

SQCHxDSCyCR.AUXOP: “0b”

SQCHxDSCyCR.ACTCODE: Rx Result Save Slot number

When SQCHxDSC00BR.DTSEL = “00b” is set, receiving payload data of 17 bytes or more is not supported. Therefore, if there is a possibility of receiving payload data of 17 bytes or more, please set SQCHxDSC00BR.DTSEL = “01b”.

After the above settings, write “1” to SQCHxSET0R.START to send the read packet.

BTA is performed after sending the read packet. When the bus right is transferred to the peripheral, the response packet is received. At this time, the Acknowledge and Error Report Packet may also be received at the same time.

The user can know the completion of Single Packet Reception by asserting SQCHxSR.ADESFIN.

The user can get an overview of the reception result by reading the RXRSSxR register with the number specified by SQCHxDSCyCR.ACTCODE, and can know the details of the previous reception result by reading the RXSR register.

If the received response packet is a Short Packet or a Long Packet with WC = 0, the setting value of SQCHxDSC00BR.DTSEL is meaningless.

Figure 34.10 shows the flow of Single Packet Reception.

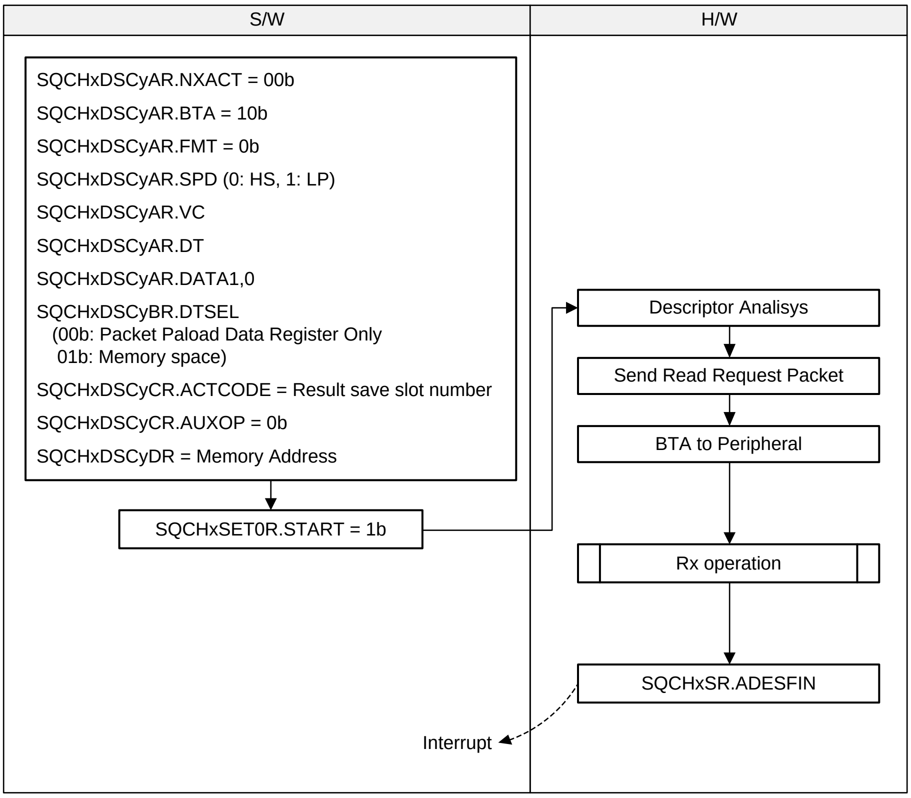


Figure 34.10 Single Packet Receive

(4) Reset Trigger Transmission

The DSI-Tx Module can send triggers using the Sequence Operation channel.

Since the DSI-Tx Module is a host, the only trigger that can be sent is a reset trigger.

After setting SQCHxDSCyAR.NXACT = “00b”, SQCHxDSCyCR.AUXOP = “1b”, and SQCHxDSCyCR.ACTCODE = “H’00”, write “1” to SQCHxSET0R.START to send a reset trigger.

The user can know the completion of reset trigger transmission by asserting SQCHxSR.ADESFIN.

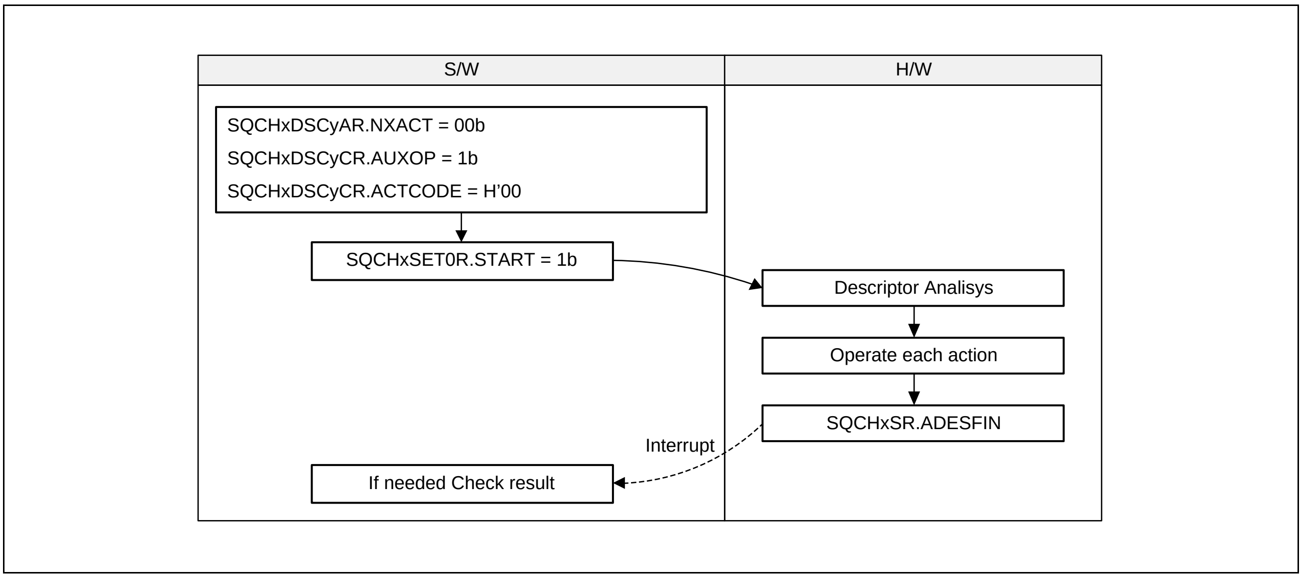


Figure 34.11 Trigger transmission

(5) Execute Operation in Sequence

The following describes the descriptor in Sequence Operation.

By writing “1” to SQCHxSET0R.START, Sequence Operation channel x will be enabled and processing will be performed from descriptor # 0.

When the processing of the corresponding descriptor is completed, the next processing follows SQCHxDSCyAR.NXACT.

SQCHxDSCyAR.NXACT= “00b”: Sequence Operation channel x stop after the processing of the corresponding descriptor is completed

SQCHxDSCyAR.NXACT= “01b”: After the processing of the corresponding descriptor is completed, the next descriptor is executed. The next descriptor after descriptor # N is descriptor # (N + 1).

(6) Receiving EoTp

The DSI-Tx Module can receive End of Transmission Packets (EoTp).

The DSI-Tx Module notifies the reception of EoTp by setting RXSR.RXEOTP to “1”.

(7) Acknowledge and Error Reporting Mechanism

The user can know the reliability of the serial bus by using “Acknowledge and Error Reporting Mechanism”.

If Peripheral detects an error on the serial bus during the period from the previous Peripheral to Host communication to this Peripheral to Host communication, Peripheral will send an “Acknowledge and Error Report” packet.

When the DSI-Tx Module receives the "Acknowledge and Error Report” packet, it sets “1” to RXRSSxR.RXAKE, RXSR.RXAKE, and SQCHxSR.RXAKE. The Virtual Channel Identifier and Error Report of the latest “Acknowledge and Error Report” packet received are stored in the AKEPLATIR register, and the integrated value is displayed in the AKEPACMSR register.

34.4.2.4 Video-Input Operation

(1) Start of Video-Input Operation

The procedure for Start of Video-Input Operation is shown in **Figure 34.12**.

1. Initialize common settings. Refer **Section 34.4.2.1(1)**.
2. Some peripheral setting may be done by Sequence operation. Since it is differed from peripheral devices, it is out of this document.
3. Confirm Video-Input clock and hsclk stable and set HS clock. Refer **Section 34.4.2.2**.
4. Set Video-Input Operation channel parameters (VICH1PPSETR, VICH1VSSETR, VICH1VPSETR,VICH1HSSETR, VICH1HPSETR)
5. Calculate VICH1SET1R.DLY value and set. Refer **Section 34.4.2.4(7)**.
6. Set “1” to VICH1SET0R.VSTART with other VICH1SET0R setting values.
7. Wait until VICH1SR.VIRDY=1.
8. Start Video signal input

VICH1SET0R, VICH1SET1R, VICH1PPSETR, VICH1VSSETR, VICH1VPSETR, VICH1HSSETR and VICH1HPSETR are prohibited to change value between VICH1SET0R.VSTART=1 and VICH1SR.RUNNING=0 after VICH1SET0R.STOP=1.

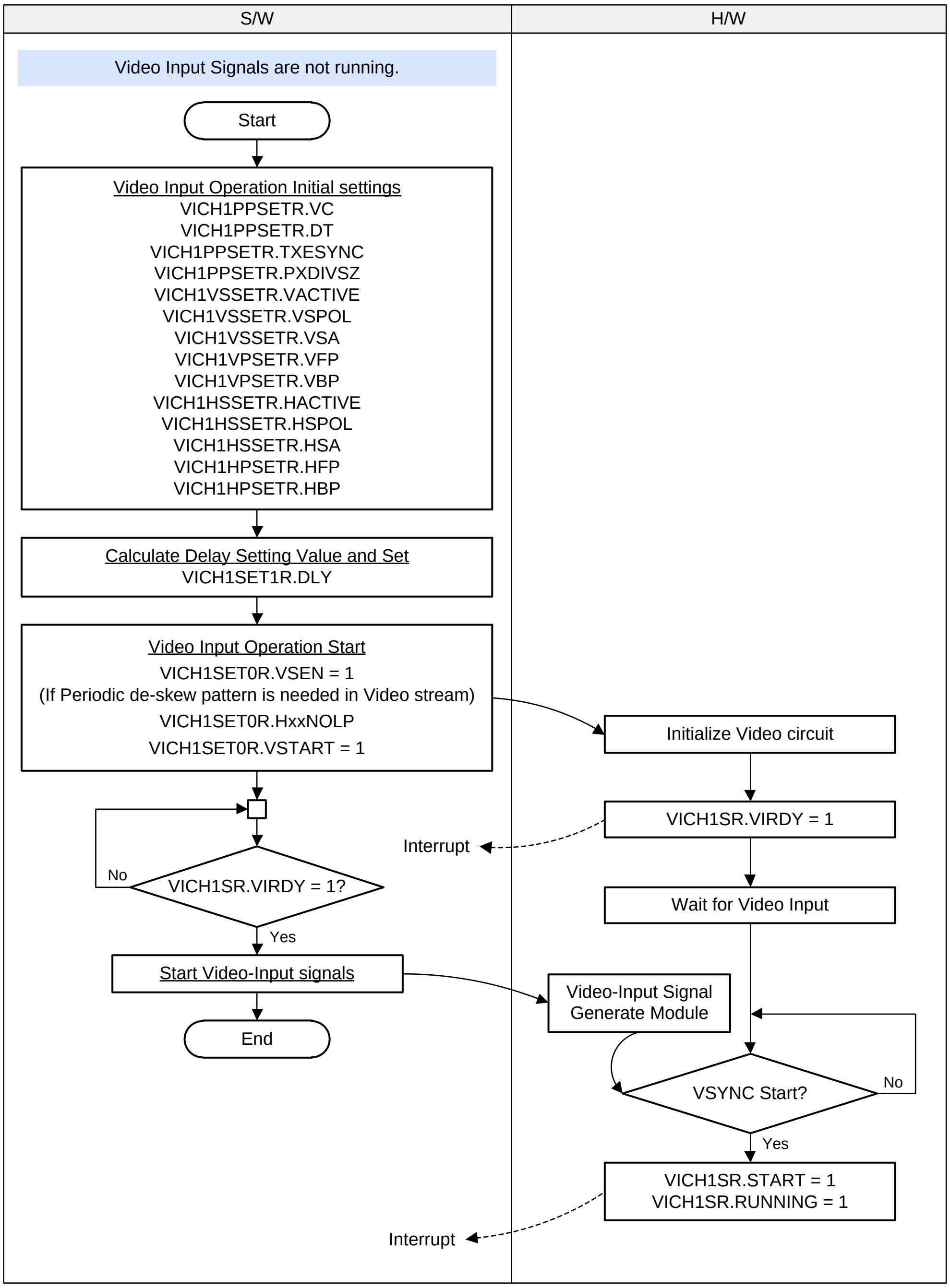


Figure 34.12 Video-Input channel Start

(2) End of Video-Input Operation

The video-input operation stop flow is shown in **Figure 34.13**. Execute the flow when VICH1SR.RUNNING = 1.

(1) VICH1SET0R.VSTPAFT (**Figure 34.13**)

Write 1 to VICH1SET0R.VSTPAFT to request the Video-Input Operation to stop. When the DSI-Tx Module detects the start of one frame (assertion of VSYNC), it stops sending video mode packets and sets VICH1SR.STOP to 1.

After checking VICH1SR.STOP, stop the video input. The video input can be stopped at any time after VICH1SR.STOP becomes 1.

After stopping the video input, wait for LINKSR.HSBUSY to become 0.

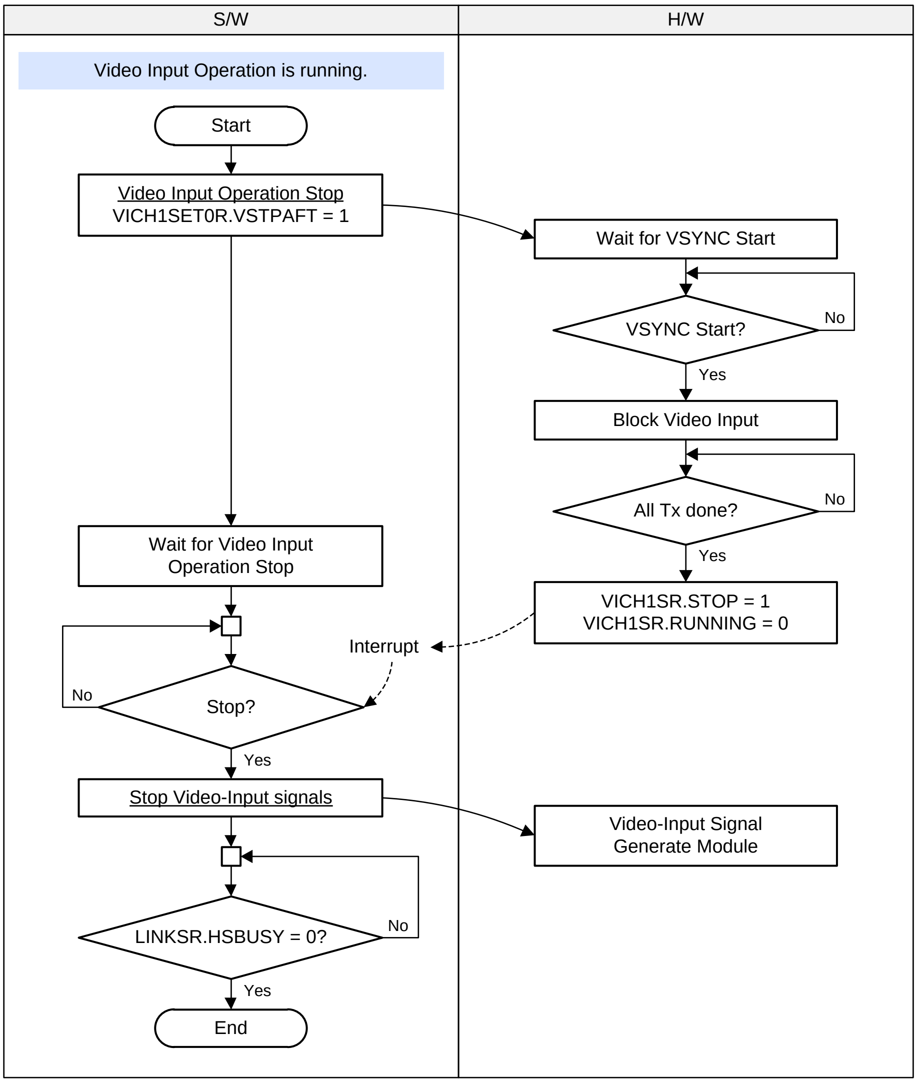


Figure 34.13 End of Video-Input Operation (VICH1SET0R.VSTPAFT)

(3) Burst Mode

The DSI-Tx Module supports Burst Mode.

Burst Mode is a mode that time-compresses pixel data.

Make the D-PHY bandwidth wider than the Video-Input bandwidth for time compression of pixel data. Video transmissions with a 1-line pixel data size (VICH1HSSETR.HACTIVE * VICH1SET1R.BPP / 8) greater than 65,535 Bytes are not supported.

(4) Non-Burst Mode

The DSI-Tx Module supports Non-Burst Mode.

Make the D-PHY bandwidth the same as the Video-Input bandwidth. Video transmissions with a 1-line pixel data size (VICH1HSSETR.HACTIVE * VICH1SET1R.BPP / 8) greater than 65,535 Bytes are not supported.

(5) LP Transition and Blanking

The DSI-Tx Module automatically transitions the Data Lane from HS to LP if both of the following two conditions are met:

If it does not transition from HS to LP, it sends a Blanking Packet with the same VC as the VC in the previous packet to maintain the HS.

- There is time to make an HS => LP => HS transition before the next HS transfer starts.
- The period is not prohibited by VICH1SET0R.xxxNOLP to transition to LP

(6) Non video packet action during Video-Input Operation Running

(a) Packet Transmission/Reception by Sequence operation

The DSI-Tx Module can send and receive packets (see **Section 34.4.2.3(2)** and **Section 34.4.2.3(3)**) using Sequence Operation while Video-Input Operation is operating*[1].

Make sure that all Sequence Operations are finished before starting the Video-Input Operation operation.

While Video-Input Operation is operating*[1], packet transmission / reception using Sequence Operation is performed only immediately after the HSE packet (VICH1PPSETR.TXESYNC = 1b) or VSS packet (VICH1PPSETR.TXESYNC = 0b) on the first horizontal line. Also, sending packets using Sequence Operation is possible only with HS (SQCHxDSCyAR.SPD = 0b). Sending with LP (SQCHxDSCyAR.SPD = 1b) is not supported. Send and receive packets that are completed within the BLLP period of the first horizontal line.

Execution of Sequence Operation processing is prohibited except for sending and receiving HS packets.

Note 1. “while Video-Input Operation is operating” here is different from VICH1SR.RUNNING. It means the period from setting VICH1SET0R.VSTART to 0 in the flow of **Figure 34.12** until VICH1SR.RUNNING becomes 0 in the flow of **Figure 34.13**. Also, please note that there is some latency from executing VICH1SET0R.VSTART to being transmitted to the inside of the module for clock transfer.

(7) Delay about Video-Input Operation

Video-Input Operation run in some latency which call in the DSI-Tx Module “Delay”. “Delay” value must set to VICH1SET1R.DLY before operation.

“Delay” value is defined by calculation using below parameters.

- Clock frequencies
  - Video channel 1 clock (MIPI_DSI_VCLK) frequency
  - DSI High-Speed Transmit Word Clock (TxWordClkHS, hsclk) frequency
- Video parameters
  - HSA period, set by VICH1HSSETR.HSA
  - HACTIVE period, set by VICH1HSSETR.HACTIVE
  - HBP period, set by VICH1HPSETR.HBP
  - HFP period, set by VICH1HPSETR.HFP
  - BPP value, shown in VICH1SET1R.BPP and derived from VICH1PPSETR.DT
- Video transmit settings
  - The number of active lanes, set by TXSETR.NUMLANEUSE
  - Transmit mode, set by VICH1PPSETR.TXESYNC
  - No LP, set by VICH1SET0R.HSANOLP, HBPNOLP, HFPNOLP
  - Clock mode, set by HSCLKSETR.HSCLKMODE
  - Clock stop time, set by CLSTPTSETR.CLKSTPT
  - Go LP and back time, set by LPTRNSTSETR.GOLPBKT
- Other parameters
  - \- Video-Input channel buffer size, shown in VICH1SET1R.CHBUFSZ

34.4.2.5 Enter/Exit of ULPS

The following describes the ULPS transition / return sequence.

The clock lane and data lane (lanes 0 to 3) can be independently transitioned / restored to ULPS.

The clock lane can make ULPS transitions when TXSETR.CLEN = 1. It is forbidden to change TXSETR.CLEN to 0 while the clock lane is ULPS.

The data lane can make ULPS transitions when TXSETR.DLEN = 1. It is forbidden to change TXSETR.DLEN to 0 while the data lane is ULPS.

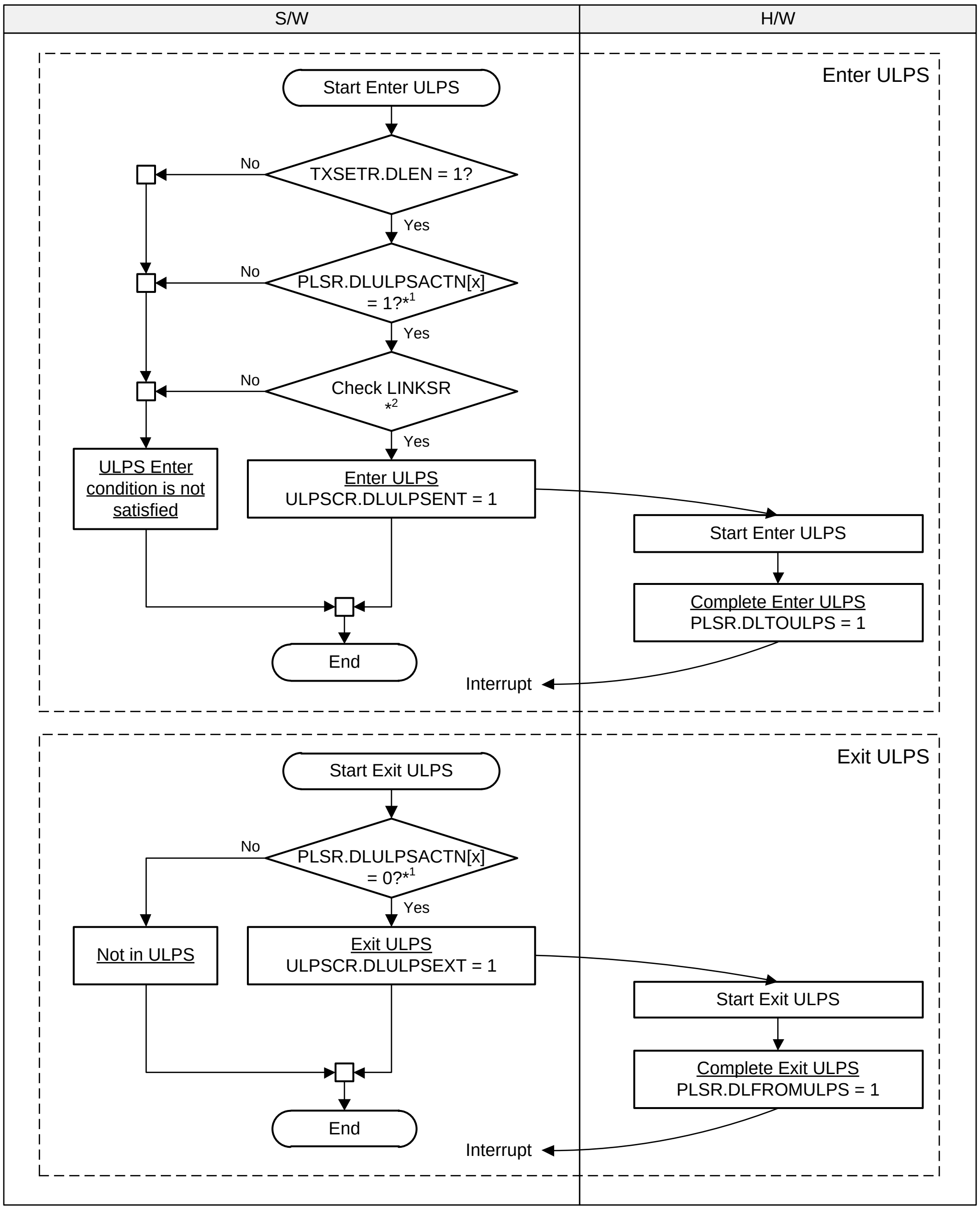
Data lanes enabled in TXSETR.NUMLANEUSE can be ULPS transitioned / restored. ULPS transition / return of data lanes is done simultaneously for all valid lanes. Only certain data lanes cannot be ULPS transitioned / restored.

When the data lane transitions to ULPS, another operation*[1] using the data lane cannot be performed until ULPS is restored.

When the clock lane transitions to ULPS, another operation*[2] using the clock lane cannot be performed until ULPS is restored.

Note 1. Sequence Operation, Video-Input Operation.

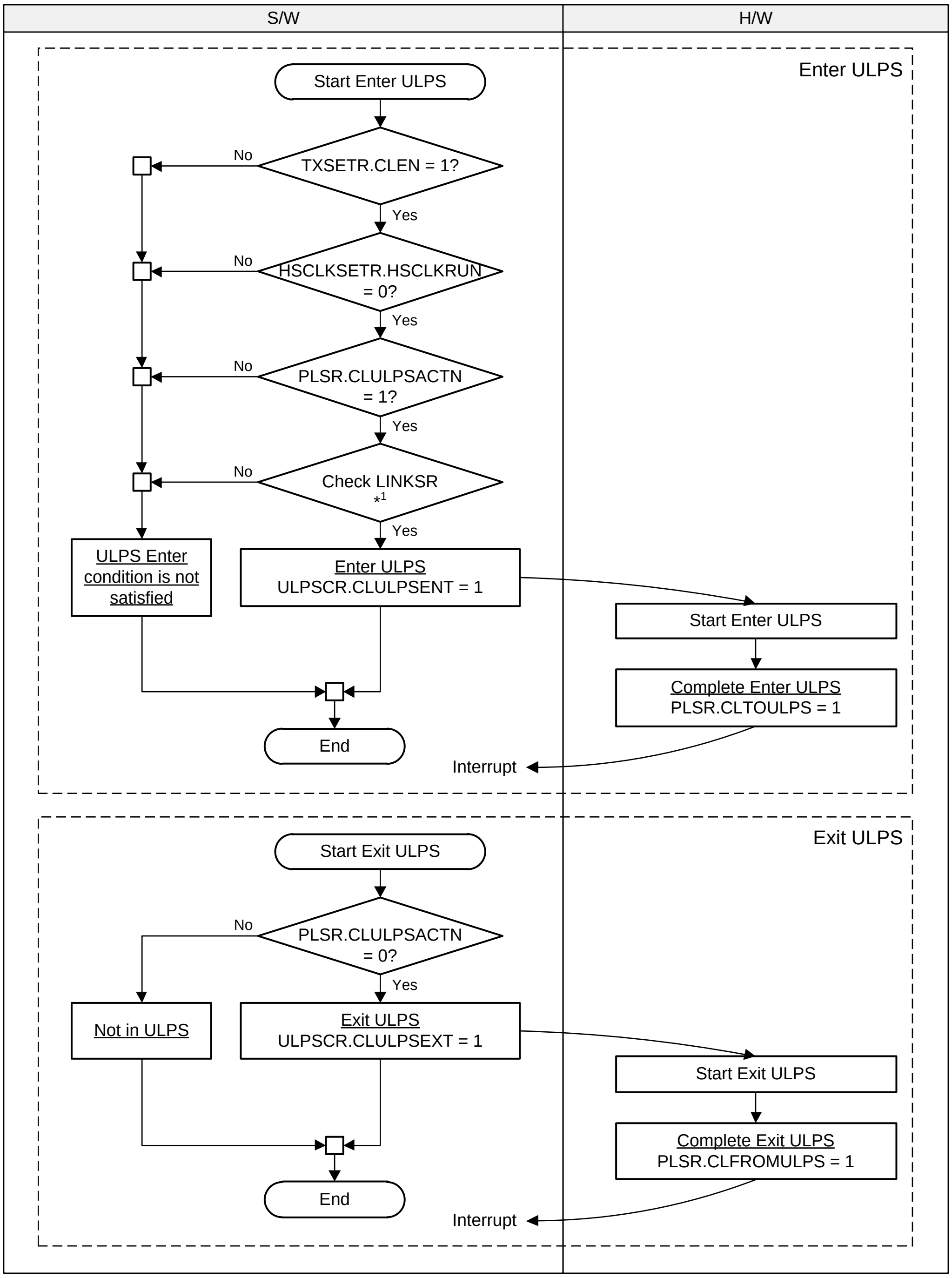
Note 2. HS Sequence Operation, Video-Input Operation.



Note 1. Check all enabled data lane which selected by TXSETR.NUMLANEUSE

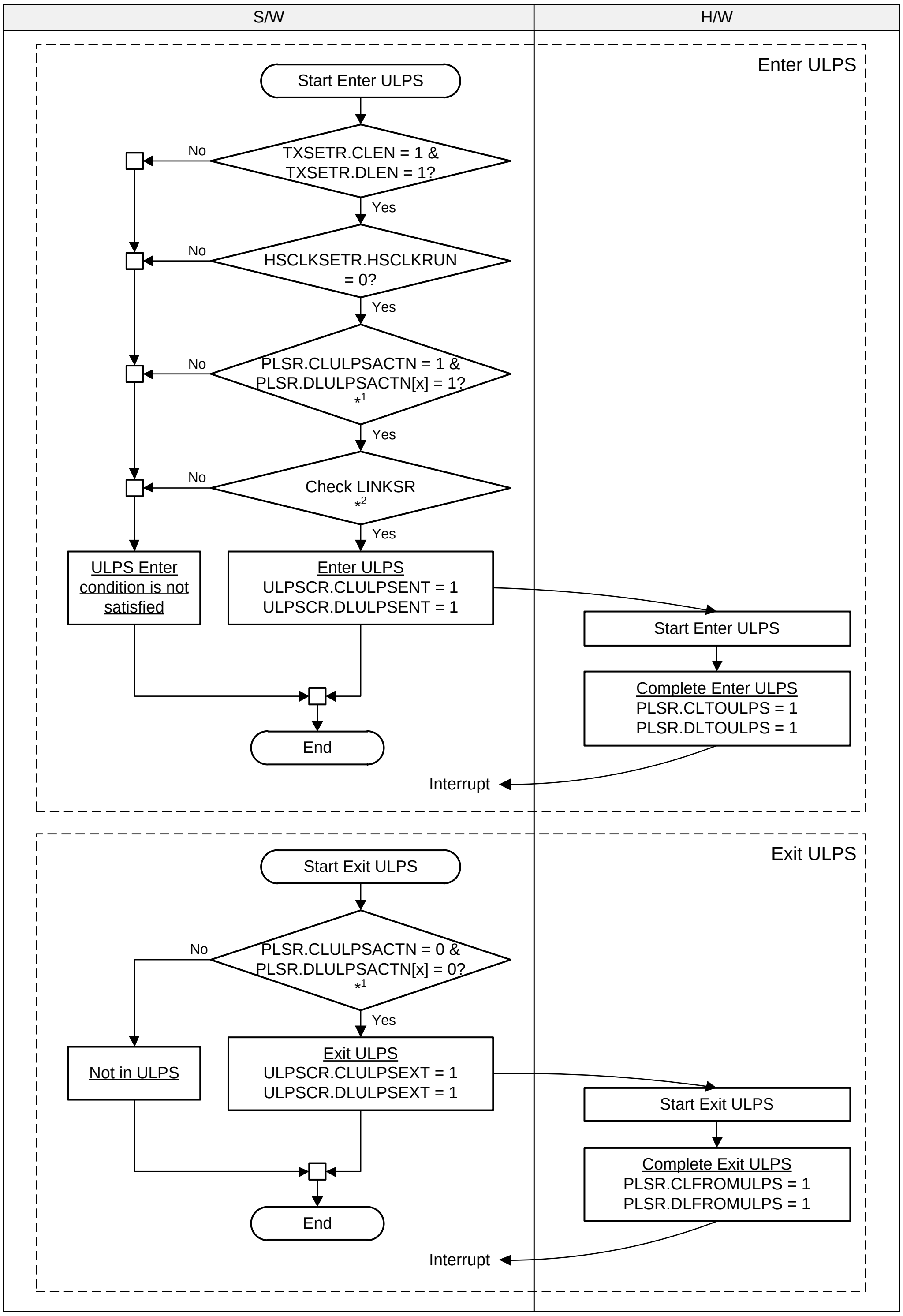
Note 2. (SQCHRUN0=0 and SQCHRUN1=0 and VICHRUN1=0 and HSBUSY=0 and LPBUSY=0)?

Figure 34.14 Enter/Exit of ULPS (Data Lane)



Note 1. (SQCHRUN1=0 and VICHRUN1=0 and HSBUSY=0)?

Figure 34.15 Enter/Exit of ULPS (Clock Lane)



Note 1. Check all enabled data lane which selected by TXSETR.NUMLANEUSE

Note 2. (SQCHRUN0=0 and SQCHRUN1=0 and VICHRUN1=0 and HSBUSY=0 and LPBUSY=0)?

Figure 34.16 Enter/Exit of ULPS (Clock and Data Lane)

34.4.2.6 Error Handling

(1) Error Detection by Peripheral

Peripheral will notify you of the detected error by “Acknowledge and Error Report” packet.

When the DSI-Tx Module receives the “Acknowledge and Error Report” packet, it displays the latest result in the AKEPLATIR register and the accumulated result in the AKEPACMSR register.

When the user receives the “Acknowledge and Error Report” packet, review and change various settings as necessary.

(2) Error Detection in D-PHY

If the D-PHY detects an error, the DSI-Tx Module sets the corresponding register flag.

The errors supported by the DSI-Tx Module are as follows. For details on the error, refer to Annex A Logical PHY-Protocol Interface Description of MIPI Alliance Specification for D-PHY Version 2.1.

- ErrEsc
- ErrSyncEsc
- ErrControl
- ErrContentionLP0
- ErrContentionLP1

If the user detects this error, perform Software Reset (**Section 34.4.2.1(2)**) and Reset Trigger Transmission (**Section 34.4.2.3(4)**) if necessary.

When an ErrContentionLP0 / ErrContentionLP1 error occurs, be sure to execute Software Reset (**Section 34.4.2.1(2)**). At that time, write 1 to RSTCR.FCETXSTP, wait for the time when Contention is resolved (equivalent to LRX-H_TO), and then write 0 to RSTCR.FCETXSTP. Also, if necessary, perform Reset Trigger Transmission (**Section 34.4.2.3(4)**).

For more information on returning from Contention, see *Section 7.2, Contention Detection and Recovery* in *the MIPI Alliance Specification* for Display Serial Interface 1.3.1.

(3) Timeout Error

The DSI-Tx Module has the timers defined in *Section 7.2.2, Contention Recovery Using Timers of the DSI standard* and the DSI-Tx Module original timers.

- Timers defined in DSI Specification
  - HS TX Timeout Error (HTX_TO)
  - LP-RX Host Processor Timeout Error (LRX-H_TO)
  - Turnaround Acknowledge Timeout Error (TA_TO)
- Original Timers
  - Rx Response Timeout Error

(a) HS TX Timeout (HTX_TO) Error

If this error occurs, perform Software Reset (**Section 34.4.2.1(2)**). In addition, perform Reset Trigger Transmission (**Section 34.4.2.3(4)**) if necessary.

If this error is detected, the setting value of HSTXTOSETR.HTXTO may be too small. When multiple channels are operating at the same time, multiple HS packets may be concatenated and transmitted continuously by HS. Set HSTXTOSETR.HTXTO to a value larger than the HS transmission period.

(b) LP-RX Host Processor Timeout (LRX-H_TO) Error

If this error occurs, perform Software Reset (**Section 34.4.2.1(2)**). In addition, perform Reset Trigger Transmission (**Section 34.4.2.3(4)**) if necessary.

If this error is detected, the setting value of LRXHTOSETR. LRXHTO may be too small.

There is a possibility of receiving multiple packets such as response packet and "Acknowledge and Error Report" packet in one reception.

Measure the time from the completion of the Bus Turnaround sequence from Host to Peripheral (LP-11 drive detection by Peripheral) to the completion of Bus Turnaround sequence from Peripheral to Host (LP-11 drive start by Host).

Check the Peripheral specifications, and set the value of DSI_LPCLK cycle count (rounded up) + X or more corresponding to the above time in the LRXHTOSETR.LRXHTO register.S

(c) Turnaround Acknowledge Timeout (TA_TO) Error

If this error occurs, perform Software Reset (**Section 34.4.2.1(2)**) and Reset Trigger Transmission (**Section 34.4.2.3(4)**) if necessary.

If this error is detected, the setting value of TATOSETR.TATO may be too small.

Measure the time from the start of the Peripheral Bus Turnaround sequence (LP-10 drive by the Host) to the completion (LP-11 drive detection by the Peripheral) from the Host.

Check the Peripheral specifications and set the value of DSI_LPCLK cycle count (rounded up) + X or more corresponding to the above time in the TATOSETR.TATO register. This error also occurs if the Peripheral does not support bidirectional communication.

(d) Rx Response Timeout Error

Refer to the Peripheral Response Timeout Error (**Section 34.4.2.6.4.1.1**) for more information.

(4) Rx Related Error in Sequence Operation

(a) Rx Packet Related Error in Sequence Operation

The following describes the received packet error detected by the DSI-Tx Module.

Detects up to one error per packet. (However, ECC 1bit Error may occur at the same time as other errors.)

Errors are determined in the following order:

Peripheral Response Timeout Error, Malformed Packet Error, ECC Error, Unexpected Packet Error, WC Error, CRC Error, Nothing Return Error

34.4.2.6.4.1.1. Peripheral Response Timeout Error

RXSR.PRESPTOERR is set to 1 if the BTA does not start receiving packets or triggers within the time set in the register after the bus right transitions from Host to Peripheral.

If this error is detected, packets and triggers will not be received until the bus right transitions from Peripheral to Host.

[Timeout time setting register]

Peripheral Response Timeout value for BTA-only is set by PRESPTOBTASETR.

Peripheral Response Timeout value for LP Read with BTA is set by PRESPTOLPSETR.PRESPTOLPR.

Peripheral Response Timeout value for LP Write with BTA is set by PRESPTOLPSETR.PRESPTOLPW.

Peripheral Response Timeout value for HS Read with BTA is set by PRESPTOHSSETR.PRESPTOHSR.

Peripheral Response Timeout value for HS Write with BTA is set by PRESPTOHSSETR.PRESPTOHSW.

If this error is detected, the register setting value may be too small, Measure the time from the completion of the Bus Turnaround sequence of Peripheral from the Host (LP-11 drive detection by Peripheral) to the completion of reception of the first byte of the packet or the completion of trigger reception. After checking the Peripheral specifications, set the value of DSI_LPCLK cycle number (rounded up) + X or more corresponding to the above time.

If this error occurs, perform the necessary processing such as resending the previous command.

34.4.2.6.4.1.2. Malformed Packet Error

If a packet less than 4 bytes is received, RXSR.MLFERR is set to 1. Since there is an error in the received data, perform necessary processing such as resending the previous command.

34.4.2.6.4.1.3. ECC Error

ECC generation is required for DSI v1.3.1 standard compliant Peripherals. However, since ECC generation is an option for “earlier revision of DSI Peripherals”, the DSI-Tx Module can select whether to enable or disable ECC checks using DSISETR.ECCEN.

When connecting to a Peripheral that does not support ECC generation, disable ECC checking by setting DSISETR.ECCEN to 0.

34.4.2.6.4.1.3.1. ECC 1bit Error

If an ECC 1bit error is detected, RXSR.ECCERR1B is set to 1 and 1bit error correction is performed.

The DSI-Tx Module automatically corrects the error and restores the normal packet header, so the subsequent packet reception processing continues.

34.4.2.6.4.1.3.2. ECC Multi-bit Error

If an ECC multi-bit error is detected, RXSR.ECCERR is set to 1 and the received packet is discarded. If this error occurs, perform the necessary processing such as resending the previous command.

34.4.2.6.4.1.4. Unexpected Packet Error

If an Unexpected Packet is received, RXSR.UEXPKTERR is set to 1.

When this error occurs, the packet header is stored in the RXRSSxR register, and the subsequent reception processing is stopped until the bus right transitions from Peripheral to Host.

The DSI-Tx Module determines the following packets as Unexpected Packets.

- Received packets in which Data Type field of the packet header is “Reserved” as described in Table23 Data Types for Peripheral-Sourced Packets of the DSI standard
- Received packets other than “Acknowledge and Error Report” packet and “EoTp” in BTA after sending non-read packet
- Received packets other than “Acknowledge and Error Report” packet and “EoTp” when the BTA is BTA only
- BTA after sending read packet receives response packet and then receives response packet again

34.4.2.6.4.1.5. WC Error

RXSR.WCERR is set to 1 if a long packet with payload data shorter than the WC value is received.

When this error occurs, there is an error in the payload data length, but the payload data is not discarded due to this error, and the packet header is stored in the RXRSSxR register.

Since there is an error in the received data, perform necessary processing such as resending the previous command.

34.4.2.6.4.1.6. CRC Error

To connect to Peripherals that do not support CRC generation, the DSI-Tx Module can enable or disable CRC checking for each Virtual Channel using DSISETR.CRCEN[3:0]. When connecting to a Peripheral that does not support CRC generation, disable the CRC check by setting the corresponding bit in DSISETR.CRCEN to 0.

When this error occurs, there is an error in the payload data, but the payload data is not discarded due to this error, and the packet header is stored in the RXRSSxR register.

If this error occurs, perform the necessary processing such as resending the previous command.

34.4.2.6.4.1.7. Nothing Return Error

RXSR.NORETERR is set to 1 if the BTA transitions the bus right from Host to Peripheral and then does not receive packets or triggers and the bus right transitions from Peripheral to Host.

If this error occurs, perform the necessary processing such as resending the command. If the error recurs, perform Software Reset (**Section 34.4.2.1(2)**) and Reset Trigger Transmission (**Section 34.4.2.3(4)**) as necessary.

(b) Rx Long Packet Data Payload Related Error in Sequence Operation

34.4.2.6.4.2.1. Maximum Return Packet Size Error

RXSR.MAXRPSZERR is set to 1 if the WC value of the received long packet is greater than the DSISETR.MRPSZ setting.

When this error occurs, the packet header is stored in the RXRSSxR register and the payload data is discarded.

Although the payload data is discarded, errors related to payload data reception (WC Error, CRC Error) are detected.

If this error occurs, set DSISETR.MRPSZ to a value greater than or equal to the value set by the Set Max Return Packet Size command.

If you still get this error despite the correct settings, consider setting DSISETR.MRPSZ to a larger value because Peripheral is returning a large packet in violation of the DSI standard.

34.4.2.6.4.2.2. Internal Bus Error

If Internal Bus Write detects an error, RXSR.IBERR is set to 1.

Review the settings of SQCHxDSCyDR.LADDR and DSISETR.MRPSZ.

34.4.2.6.4.2.3. Rx Buffer Overflow Error

If the following Rx Buffer Overflow Error is detected, RXSR.RXOVFERR is set to 1.

In the case of SQCHxDSCyBR.DTSEL=00b Packet Payload Data register Mode:

- The Rx Payload Data is over 16 Byte length

In the case of SQCHxDSCyBR.DTSEL=01b Long Packet data use memory space Mode:

- RAM Overflow for the LP Rx is detected

If this error occurs, please execute the following.

In the case of Data register Mode:

- Change to use memory space Mode
- Change peripheral device’s Maximum Return Packet Size as not over 16 Byte

In the case of memory space Mode:

- Change peripheral device’s Maximum Return Packet Size smaller
- Make Internal Bus faster

(5) Tx Packet Related Error in Sequence Operation

(a) Internal Bus Error

If Internal Bus Read detects an error, SQCHxSR.TXIBERR is set to 1. Review the settings of SQCHxDSCyAR.DATA1 / 0 and SQCHxDSCyDR.LADDR.

(b) Packet too big Error

SQCH1SR.PKTBIGERR is set to 1 if the Sequence Operation packet is of a size that cannot be sent within the BLLP period of the first horizontal line of the Video-Input Operation.

If this error occurs, the Sequence Operation packet will not be sent until the Video-Input Operation is completed.

(6) Error in Video-Input operation

(a) Video-Input Buffer Overflow Error

VICH1SR.VBUFOVF is set to 1 if Overflow of Video-Input Buffer is detected.

If this error occurs, perform Software Reset (**Section 34.4.2.1(2)**). Also, take necessary measures such as increasing the frequency of TxWordClkHS (hsclk) and decreasing the VICH1SET1R.DLY value. In addition, perform Reset Trigger Transmission (**Section 34.4.2.3(4)**) if necessary.

(b) Video-Input Buffer Underflow Error

If the Video-Input Buffer Underflow is detected, VICH1SR.VBUFUDF is set to 1.

If this error occurs, perform Software Reset (**Section 34.4.2.1(2)**). Also, take necessary measures such as lowering the frequency of TxWordClkHS (hsclk) and increasing the VICH1SET1R.DLY value. In addition, perform Reset Trigger Transmission (**Section 34.4.2.3(4)**) if necessary.

(c) Timing Error

VICH1SR.TIMERR is set to 1 if no video packets (including Sync Event packets) were sent during the Video-Input Operation at the intended timing.

If this error occurs, perform Software Reset (**Section 34.4.2.1(2)**). Also, take necessary measures such as reviewing the setting value of LPTRNSTSETR.GOLPBKT. In addition, perform Reset Trigger Transmission (**Section 34.4.2.3(4)**) if necessary.

34.4.3 Interface

34.4.3.1 Video-Input Operation

(1) Video Input Timing

Video-Input Timing and Video-Input Parameters are described in **Figure 34.17** and **Table 34.8**.

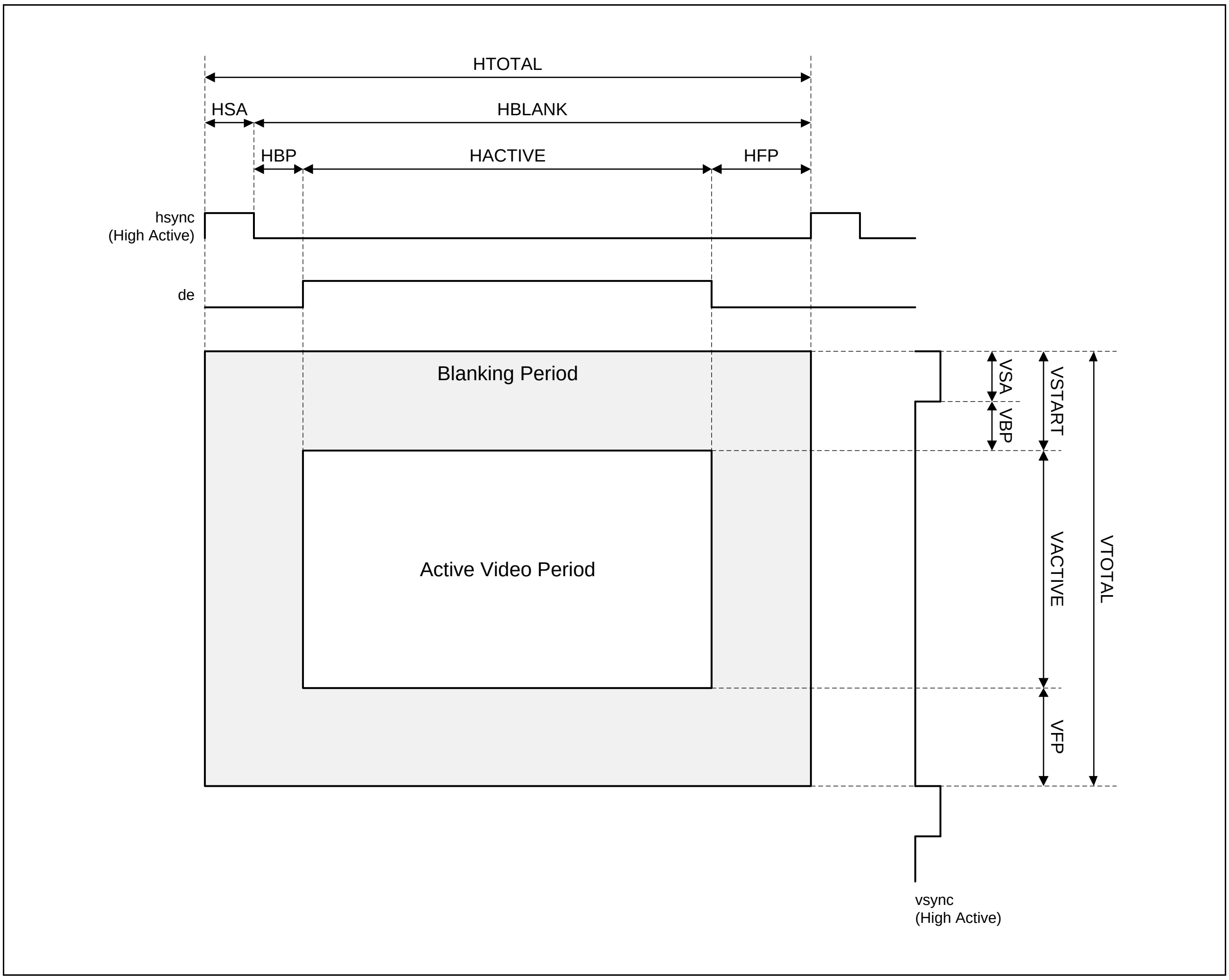


Figure 34.17 Video input Timing

Table 34.8 Configurations for Video Input interface (1/2)

- Common Rules for all mode

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>All</td><td>All parameters related to Input Video should be set to the value of not less than 1.
Example for “All parameters”: HSA, HBP, HACTIVE, HFP, VSA, VBP, VACTIVE, VFP</td></tr>
<tr><td>HACTIVE</td><td><table>
<tr><th>VICH1PPSETR.DT</th><th>VICH1HSSETR.HACTIVE</th></tr>
<tr><td>H’1E: Packed Pixel Stream, 18bit RGB</td><td>This value should be the multiple of 4.</td></tr>
<tr><td>H’2E: Loosely Packed Pixel Stream, 18bit RGB
H’3E: Packed Pixel Stream, 24bit RGB</td><td>No Limitation</td></tr>
</table></td></tr>
<tr><td>HFP</td><td>Rounddown (HFP*BPP/ 8, 0) >= 12, (12 = means Pixel Packet Header & footer 6 + Blanking Packet Header & footer 6)</td></tr>
<tr><td>HTOTAL</td><td><table>
<tr><th>VICH1PPSETR.DT</th><th>VICH1HSSETR.HSA +
VICH1HPSETR.HBP +
VICH1HSSETR.HACTIVE +
VICH1HPSETR.HFP</th></tr>
<tr><td>H’1E: Packed Pixel Stream, 18bit RGB</td><td>This value should be the multiple of 4.</td></tr>
<tr><td>H’2E: Loosely Packed Pixel Stream, 18bit RGB
H’3E: Packed Pixel Stream, 24bit RGB</td><td>No Limitation</td></tr>
</table></td></tr>
</table>

- Additional Rules for non-burst Sync Pulse mode

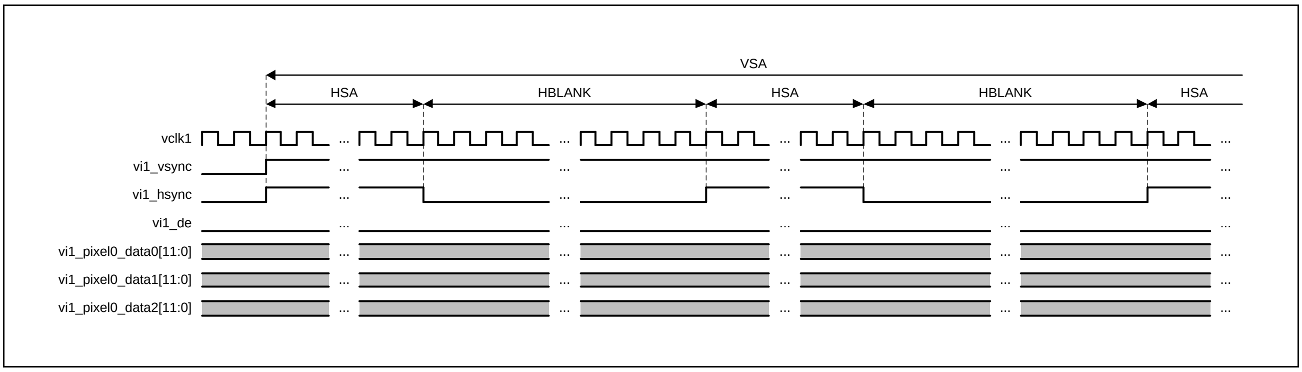
Parameter	Description
Sequence Operation Packet	When a packet by sequence operation is transmitted during Video Opearation (VICH1SR.RUNNING=1), the period of (HBP+HACTIVE+HFP) should be set the value longer than sum of below periods. ● Period for the Tx Sync Event packet (4 Byte) ● Period for the packet by sequence operation ● Period for the Header and CRC of a Blanking Packet (6 Byte)
HBP+HACTIVE+HFP	Transition of HS-LP-HS is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of (HBP+HACTIVE+HFP) should be set to the value within the range described below. (HBP+HACTIVE+HFP) > (period for the Tx Sync Event Packet) + (period for transition time as HS-LP-HS)
HSA	Transmission of Blanking Packet is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of HSA should be set to the value within the range described below. (HSA*BPP)/8 >= 10 Notice, HSA should be set to the value not less than minimum value of (period for the Tx Sync Event Packet) + (period for the Tx Blanking Packet).
HBP	Transmission of Blanking Packet is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of HPB should be set to the value within the range described below. (HBP*BPP)/8 >= 10 Notice, HPB should be set to the value not less than minimum value of (period for the Tx Sync Event Packet) + (period for the Tx Blanking Packet).

Table 34.8 Configurations for Video Input interface (2/2)

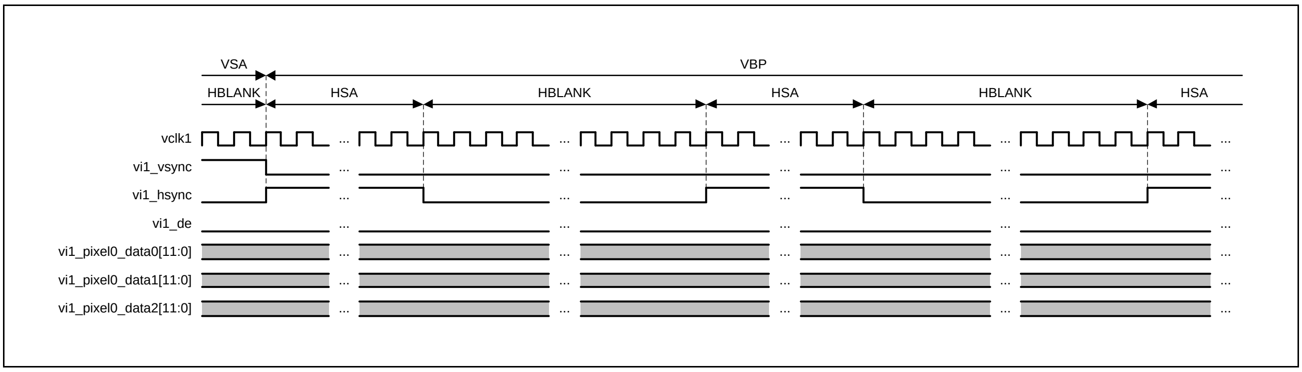
- Additional Rules for non-burst Sync-Event mode and Burst mode

Parameter	Description
Sequence Operation Packet	When a packet by sequence operation is transmitted during Video Operation (VICH1SR.RUNNING=1), the period of HTOTAL should be set the value longer than sum of below periods. ● Period for the Tx Sync Event packet (4 Byte) ● Period for the packet by sequence operation ● Period for the Header and CRC of a Blanking Packet (6 Byte)
HTOTAL	Transition of HS-LP-HS is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of HTOTAL should be set to the value within the range described below. HTOTAL > (period for the Tx Sync Event Packet) +(period for transition time as HS-LPHS)
HSA+HBP	Transmission of Blanking Packet is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of (HSA+HBP) should be set to the value within the range described below. ((HSA+HBP)*BPP)/8 >= 10 Notice, HSA+HBP should be set to the value not less than minimum value of (period for the Tx Sync Event Packet) + (period for the Tx Blanking Packet).

(1) VSA Start (and VFP End)

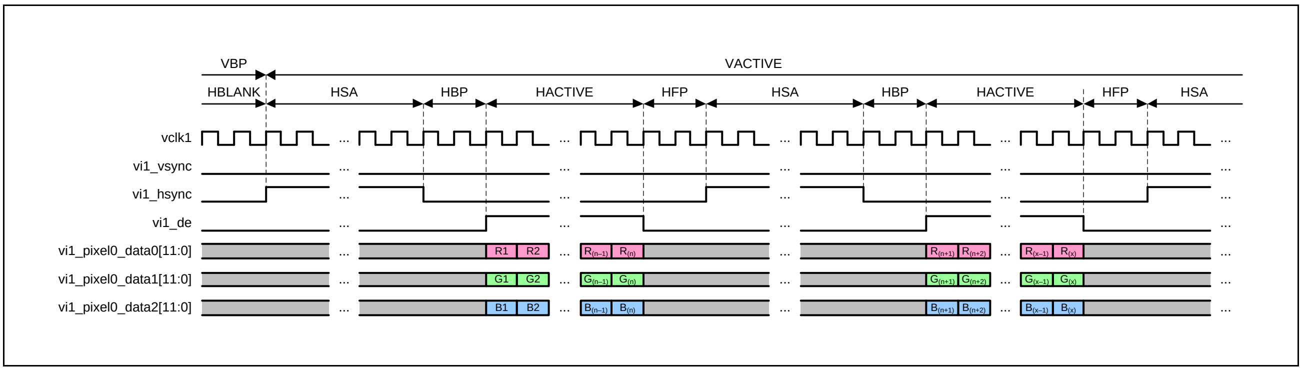


(2) VSA End and VBP Start

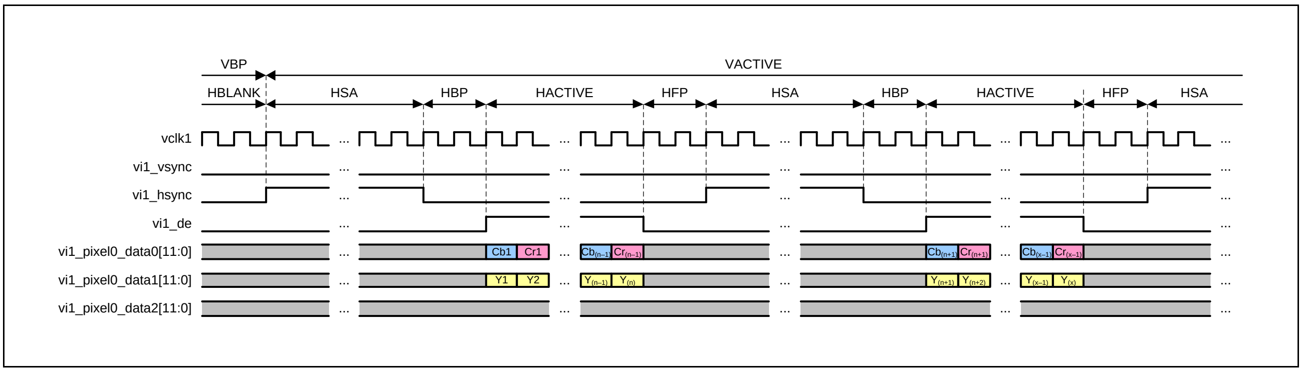


(3) VBP End and VACTIVE Start

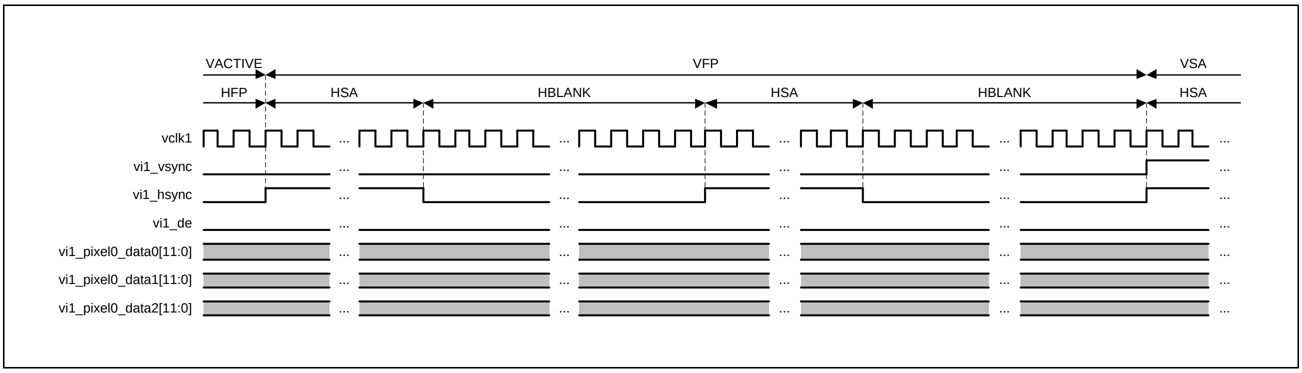
(a) RGB



(b) YCbCr 4: 2: 2



(4) VACTIVE End and VFP Start



34.4.3.2 Video Input Start, End, Change Setting

(1) Video Input Start
Video input must be started after Video-Input Operation Start sequence. It must be start from V-Sync start edge.

(2) Video Input End
End of Video Input must follow sequence of **Section 34.4.2.4(2), End of Video-Input Operation**.

(3) Change of Video Input Setting
Changing of Video-Input setting must be done during Video-Input operation is not running.

34.4.3.3 Clock Frequencies

Video clock and DSI HS Byte clock must follow the relationship.

(1) Burst Mode

Video clock Frequency [Hz] * Video Pixel Bit Depth (BPP) [bit]

<= DSI HS Byte clock Frequency[Hz] * 8[bit] * Number of DSI HS Data Lane

In the case of Video-Input bandwidth equals DSI output bandwidth, it will be indistinguishable from non-burst sync-event mode.

(2) Non-Burst Mode

Video clock Frequency [Hz] * Video Pixel Bit Depth (BPP) [bit]

= DSI HS Byte clock Frequency[Hz] * 8[bit] * Number of DSI HS Data Lane

34.5 Usage Note

34.5.1 Restriction and Notes

In this chapter, restriction and note of the DSI-Tx Module are described.

Table 34.9 Attention and Restriction List

Items	Restriction and Notes
TxWordClkHS	Each Lanes TxWordClkHS must be synchronized. The DSI-Tx Module uses Data Lane0 TxWordClkHS as hsclk.
Unused input signals	All unused input signals must be input 0, except following signals. Following signals must be input 1, when they are not used. ppi_dl1_ulpsactivenot ppi_dl2_ulpsactivenot ppi_dl3_ulpsactivenot
TxReadyHS assert	D-PHY must be asserted TxReadyHS within ($2^{\mathrm{RELDSI2TX\_LNK\_RAM\_WORD\_DEPTH}}$ – 2) hsclk cycles after TxRequestHS asserted.

35. Camera Data Receiving Unit (CRU)

The CRU consists of a MIPI CSI-2 block and an Image Processing block.

(1) MIPI CSI-2

This block can receive signals conforming to the MIPI CSI-2 standard, extracts video signal from various packets, and sends them to Image Processing block in the subsequent stage.

(2) Image Processing

This block can receive video data received from the external Digital Parallel Interface or MIPI CSI-2 block, and perform appropriate image processing for each. The image-processed data is temporarily stored in the FIFO and transferred to an external memory.

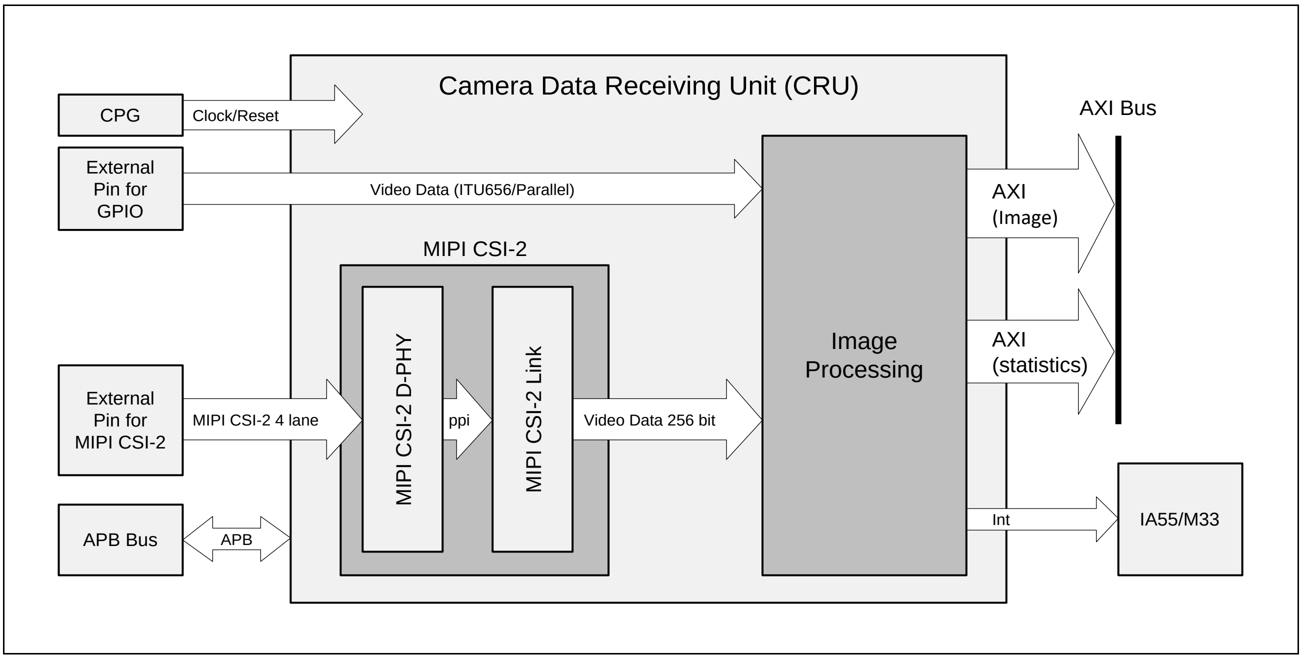


Figure 35.1 Block Diagram of CRU

35.1 MIPI CSI-2

35.1.1 Overview

The MIPI CSI-2 is an MIPI Camera Serial Interface 2 receiver module.

This module supports MIPI CSI-2 V2.1 and MIPI D-PHY V2.1 (80 Mbps ~ 1500 Mbps).

The image signal received by the MIPI-CSI2 is output to the image processing module.

35.1.2 Features

This module supports the following functions.

1) Input Image Size

- Maximum image size: 5 M pixels
- Minimum image size: QVGA (320 × 240) ＝ 76.8 K pixels
- Maximum number of valid pixels in the horizontal direction: 2800 pixels
- Maximum number of valid pixels in the vertical direction: 4095 lines

2) Compliant with Specification for Camera Serial Interface 2 (CSI-2)^{SM}

3) Support 1/2/4 lanes

4) Supported Input Data Format is described in **Table 35.12**.

5) Support Data De-Scrambling

6) Generic Short Packet FIFO with the depth of 16

7) Support Latency Reduction and Transport Efficiency (LRTE)

8) ECC 1bit error correction and 2bit error detection in packet header

9) CRC check for payload data

10) Support 4 Virtual Channel

11) Support Data Interleaving

- Data Type Interleaving
  - Packet Level Interleaved Data Transmission
  - Frame Level Interleaved Data Transmission
- Virtual Channel Identifier Interleaving

12) Receiver Error Detection and Report

- D-PHY Level Errors
- Packed Level Errors
- Protocol Decoding Level Errors

13) Skew Adjustment for D-PHY

14) MIPI CSI-2 Input Bandwidth Limitation

- CRU can receive signals which satisfied both of **Table 35.1** and **Table 35.2**.

15) Blanking period

- Minimum invalid period between horizontal lines: 412ns.

Table 35.1 Supported Image Format and Maximum Transfer Rate

Number of Lanes	Transfer Data Rate per Lane [M bps / Lane]		Total Transfer Data Rate [M bps]	
	MIN	MAX	MIN	MAX
1 lane	80	1500	80	1500
2 Lanes	80	1500	160	3000
4 Lanes	80	1500	320	6000

Table 35.2 Supported Image Format and Maximum Transfer Rate

No.	Input Image Format	MIPI CSI-2 Maximum Transfer Ratio [Mbps]
1	YCbCr/YUV422 8-bit	4256
2	YCbCr/YUV422 10-bit	5238
3	RGB565	4256
4	RGB666	4540
5	RGB888	6000
6	RAW8	2128
7	RAW10	2619
8	RAW12	3095
9	RAW14	3584
10	RAW16	4000
11	Others	2128

35.1.3 Block Diagram

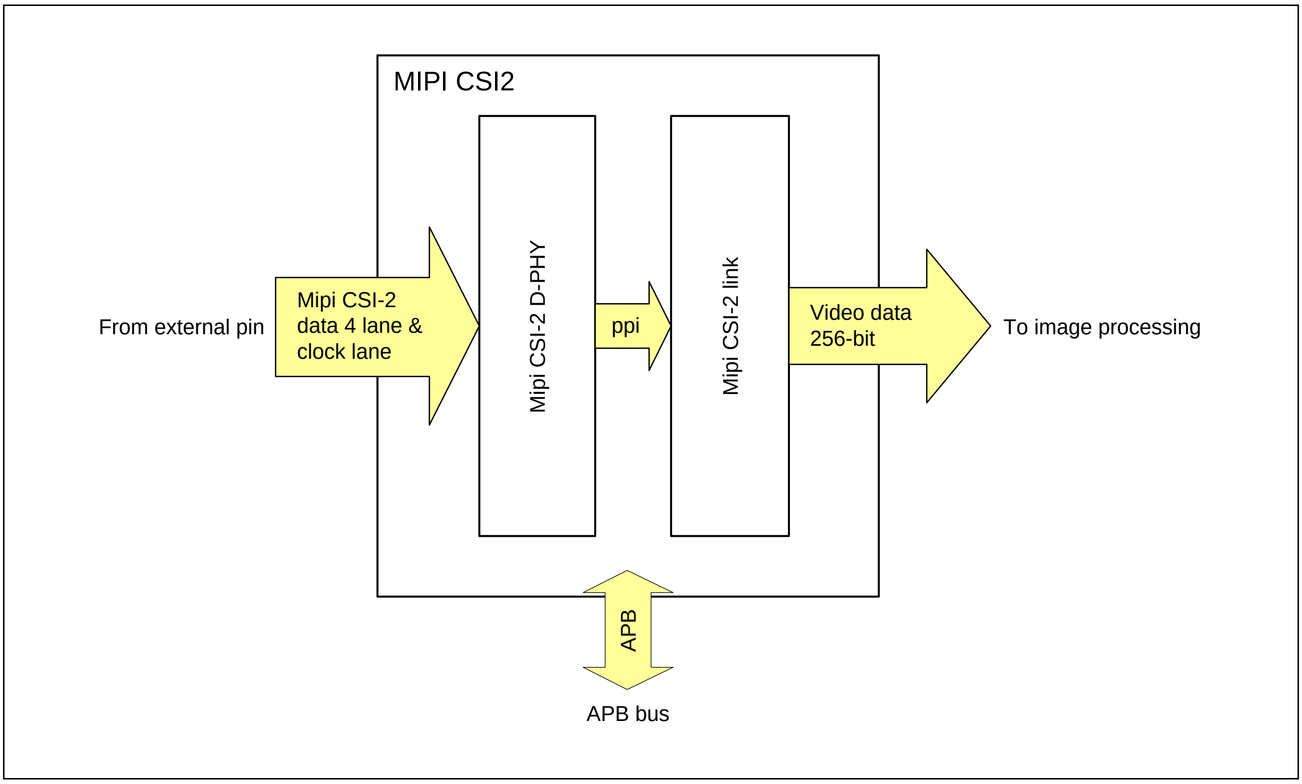


Figure 35.2 Block Diagram of MIPI CSI-2

35.1.4 External Pins

Table 35.3 shows the pin configuration.

Table 35.3 External Pins

Pin Name	Input/Output	Function
CSI_CLKP	Input	MIPI CSI-2 Rx clock input (positive)
CSI_CLKN	Input	MIPI CSI-2 Rx clock input (negative)
CSI_DATA0_P	Input	MIPI CSI-2 Rx serial data inputs lane0 (positive)
CSI_DATA0_N	Input	MIPI CSI-2 Rx serial data inputs lane0 (negative)
CSI_DATA1_P	Input	MIPI CSI-2 Rx serial data inputs lane1 (positive)
CSI_DATA1_N	Input	MIPI CSI-2 Rx serial data inputs lane1 (negative)
CSI_DATA2_P	Input	MIPI CSI-2 Rx serial data inputs lane2 (positive)
CSI_DATA2_N	Input	MIPI CSI-2 Rx serial data inputs lane2 (negative)
CSI_DATA3_P	Input	MIPI CSI-2 Rx serial data inputs lane3 (positive)
CSI_DATA3_N	Input	MIPI CSI-2 Rx serial data inputs lane3 (negative)

35.1.5 Register Configuration

Module and base address are shown below.

Module Name	Address
MIPI CSI-2 LINK	H’0_1083_0400 (Cortex-A55 Address Space) H’4083_0400 (Cortex-M33 Address Space Non-Secure) H’5083_0400 (Cortex-M33 Address Space Secure)
MIPI CSI-2 D-PHY	H’0_1083_0800 (Cortex-A55 Address Space) H’4083_0800 (Cortex-M33 Address Space Non-Secure) H’5083_0800 (Cortex-M33 Address Space Secure)

Table 35.4 Register Configuration of MIPI CSI-2 LINK (1/2)

Name	Abbreviation	Offset Address	Initial Value
Module Configuration Register	CSI2nMCG	H’000	H’0110_0400
Module Control Register 0	CSI2nMCT0	H’010	H’0200_0004
Reserved	—	H’014	H’0000_0003
Module Control Register 2	CSI2nMCT2	H’018	H’0000_0000
Module Control Register 3	CSI2nMCT3	H’01C	H’0000_0000
Reset Control Register	CSI2nRTCT	H’028	H’0000_0000
Reset Status Register	CSI2nRTST	H’02C	H’0000_0000
Reserved	—	H’030	H’7654_3210
EPD Option Control Register	CSI2nEPCT	H’040	H’0000_0000
Module Interrupt Status Register	CSI2nMIST	H’050	H’0000_0000
Receive Data Type Enable Low Register	CSI2nDTEL	H’060	H’0000_000F
Receive Data Type Enable High Register	CSI2nDTEH	H’064	H’0000_0000
Receive Status Register	CSI2nRXST	H’070	H’0000_0000
Receive Status Clear Register	CSI2nRXSC	H’074	H’0000_0000
Receive Interrupt Enable Register	CSI2nRXIE	H’078	H’0000_0000
Data Lane 0 Status Register	CSI2nDLST0	H’080	H’0000_0000
Data Lane 0 Status Clear Register	CSI2nDLSC0	H’084	H’0000_0000
Data Lane 0 Interrupt Enable Register	CSI2nDLIE0	H’088	H’0000_0000
Data Lane 1 Status Register	CSI2nDLST1	H’090	H’0000_0000
Data Lane 1 Status Clear Register	CSI2nDLSC1	H’094	H’0000_0000
Data Lane 1 Interrupt Enable Register	CSI2nDLIE1	H’098	H’0000_0000
Data Lane 2 Status Register	CSI2nDLST2	H’0A0	H’0000_0000
Data Lane 2 Status Clear Register	CSI2nDLSC2	H’0A4	H’0000_0000
Data Lane 2 Interrupt Enable Register	CSI2nDLIE2	H’0A8	H’0000_0000
Data Lane 3 Status Register	CSI2nDLST3	H’0B0	H’0000_0000
Data Lane 3 Status Clear Register	CSI2nDLSC3	H’0B4	H’0000_0000
Data Lane 3 Interrupt Enable Register	CSI2nDLIE3	H’0B8	H’0000_0000
Virtual Channel 0 Status Register	CSI2nVCST0	H’100	H’0000_0000
Virtual Channel 0 Status Clear Register	CSI2nVCSC0	H’104	H’0000_0000
Virtual Channel 0 Interrupt Enable Register	CSI2nVCIE0	H’108	H’0000_0000
Virtual Channel 1 Status Register	CSI2nVCST1	H’110	H’0000_0000
Virtual Channel 1 Status Clear Register	CSI2nVCSC1	H’114	H’0000_0000
Virtual Channel 1 Interrupt Enable Register	CSI2nVCIE1	H’118	H’0000_0000
Virtual Channel 2 Status Register	CSI2nVCST2	H’120	H’0000_0000

Table 35.4 Register Configuration of MIPI CSI-2 LINK (2/2)

Name	Abbreviation	Offset Address	Initial Value
Virtual Channel 2 Status Clear Register	CSI2nVCSC2	H’124	H’0000_0000
Virtual Channel 2 Interrupt Enable Register	CSI2nVCIE2	H’128	H’0000_0000
Virtual Channel 3 Status Register	CSI2nVCST3	H’130	H’0000_0000
Virtual Channel 3 Status Clear Register	CSI2nVCSC3	H’134	H’0000_0000
Virtual Channel 3 Interrupt Enable Register	CSI2nVCIE3	H’138	H’0000_0000
Power Management Status Register	CSI2nPMST	H’200	H’0000_0000
Power Management Status Clear Register	CSI2nPMSC	H’204	H’0000_0000
Power Management Interrupt Enable Register	CSI2nPMIE	H’208	H’0000_0000
Generic Short Packet Control Register	CSI2nGSCT	H’280	H’0001_0000
Generic Short Packet Status Register	CSI2nGSST	H’284	H’0000_0000
Generic Short Packet Status Clear Register	CSI2nGSSC	H’288	H’0000_0000
Generic Short Packet Interrupt Enable Register	CSI2nGSIE	H’28C	H’0000_0000
Generic Short Packet Register	CSI2nGSHT	H’290	H’0008_0000
Generic Short Packet Information Update Register	CSI2nGSIU	H’294	H’0000_0000

Note: Addresses other than above are reserved. Operation is not guaranteed if accessed them.

Table 35.5 Register Configuration of MIPI CSI-2 D-PHY

Name	Abbreviation	Offset Address	Initial Value
D-PHY Control Register 0	CSIDPHYCTRL0	H’00	H’0001_0004
D-PHY Timing Register 0	CSIDPHYTIM0	H’04	H’0100_2710
D-PHY Timing Register 1	CSIDPHYTIM1	H’08	H’0403_0909
D-PHY Skew Adjustment Function	CSIDPHYSKW0	H’60	H’0000_0000

Note: Addresses other than above are reserved. Operation is not guaranteed if accessed them.

35.1.6 Register Descriptions

Access the following registers only in units of 32 bits.

Table 35.6 lists the attributes of the MIPI CSI-2 D-PHY and MIPI CSI-2 LINK registers.

Table 35.6 Attributes of MIPI CSI-2 D-PHY and MIPI CSI-2 LINK Registers

Attribute	Write	Read
R	Disabled	Enabled
RW	Enabled	Enabled
R0W	Enabled	Enabled (always read as 0)

35.1.6.1 Module Configuration Register (CSI2nMCG)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GSNM							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SDLN				—	—	—	—	VER			
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	1b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	GSNM	H'10	R	Indicate the number of stages of the generic short packet FIFO (16 stages).
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	SDLN	H'4	R	Indicate the supported maximum number of data lanes. H'4: Operable with four lanes, two lanes or one lane.
7 to 4	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	VER	H'0	R	Indicate the version of this core.

35.1.6.2 Module Control Register 0 (CSI2nMCT0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	LFSREN	ECCV13	—	—	—	—	RVMD	—	EDMD	ZLMD
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	RW	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	VDLN			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	LFSREN	1b	RW	Enables or disables de-scrambling. 0b: Disables de-scrambling. 1b: Enables de-scrambling.
24	ECCV13	0b	RW	CSI-2 specification-compliant mode of ECC checking 0b: 26 bits to be checked for ECC 1b: 24 bits to be checked for ECC It is prohibited to set this bit to 1b when receiving four or more virtual channels.
23 to 21	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19	RVMD	0b	RW	For a packet whose Data Type is H'38 to H'3F (“Reserved” in Table 10 Data Type Classes in the CSI-2 Specification), set the receive operation mode to 0. 0b: Discards the data (notifies an ErrID). 1b: Receives data as a long packet (outputs data to the image processing module when the corresponding bit in CSI2nDTEH is 1, or discards the data and notifies an ErrID when the corresponding bit in CSI2nDTEH is 0),
18	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	EDMD	0b	RW	Notifies ErrFrameData when an ECC 2-bit error or a packet of less than four bytes is received between FS and FE. 0b: Does not notify ErrFrameData. 1b: Notifies ErrFrameData.
16	ZLMD	0b	RW	Sets whether to output a long packet whose word count is 0 to the image processing module. Set this bit to 0. 0b: Output 1b: No output
15 to 4	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	VDLN	H'4	RW	Set the number of used data lanes. Change this field when CSI2nMCT3.RXEN = 0. A value equal to or less than the CSI2nMCG.SDLN value can be specified. H'1: Operation with one lane H'2: Operation with two lanes H'4: Operation with four lanes Others: Setting prohibited

35.1.6.3 Module Control Register 2 (CSI2nMCT2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FRRSKW								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	FRRCLK								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24 to 16	FRRSKW	H'000	RW	Internal setting parameter Set the following value by using the ROUNDDOWN function that rounds down the decimal places. (ROUNDDOWN((3 * vclk frequency/hsclk1 frequency), 0) + 1) where vclk = 266 MHz, hsclk1 = transfer rate/8 Example: Set 7 when the vclk frequency is 266 MHz and the transfer rate is 1 Gbps. ROUNDDOWN((3 * 266/(1000/8)), 0) + 1 = 7
15 to 9	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8 to 0	FRRCLK	H'000	RW	Parameter for the IP to determine whether packets have been received if RxWordClkHS stops at the timing when the IP cannot recognize that RxValidHS is low. Set the following value by using the ROUNDDOWN function that rounds down the decimal places. (ROUNDDOWN((1.5 * vclk frequency/hsclk1 frequency), 0) + 1) where vclk = 266 MHz, hsclk1 = transfer rate/8 Example: Set 4 when the vclk frequency is 266 MHz and the transfer rate is 1 Gbps. ROUNDDOWN((1.5 * 266/(1000/8)), 0) + 1 = 4

35.1.6.4 Module Control Register 3 (CSI2nMCT3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RXEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	RXEN	0b	RW	Enables data reception in the PPI. The value of this register is reflected in ppi_dl[1:4]_enable and ppi_cl_enable for the lane(s) which is enabled by CSI2nMCT0.VDLN. 0b: Disables reception. 1b: Enables reception.

35.1.6.5 Reset Control Register (CSI2nRTCT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VSRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	VSRST	0b	R0W	Writing 1 to this bit resets the internal LINK signal by a software reset.

35.1.6.6 Reset Status Register (CSI2nRTST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VSRSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	VSRSTS	0b	R	Indicates the execution status of the software reset generated by CSI2nRTCT.VSRST. 0b: Not during a reset 1b: During a reset

35.1.6.7 EPD Option Control Register (CSI2nEPCT)

This register supports the TX_REG_CSI_EPD_EN_SSP register and the TX_REG_CSI_EPD_OP_SLP register that are defined in *Section 9.11.1.2.3, D-PHY EPD Specifications (for EPD Options 1 and 2)* in *the MIPI CSI-2 V2.1 Specification*.

Set the same value as the one set in this register in the CSI-2 transmitter.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EPDEN	SSP														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDOP	SLP														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>31</td><td>EPDEN</td><td>0b</td><td>RW</td><td>Enables the EPD.
0b: Disables the EPD.
1b: Enables the EPD.</td></tr>
<tr><td>30 to 16</td><td>SSP</td><td>All 0</td><td>RW</td><td>Specify the number of spacers inserted after a short packet.
This field is valid when EPDEN = 1.
The following shows the specifiable minimum number of spacers when EPDEN = 1.
Setting a value less than the minimum value is prohibited.
<table>
<tr><th>EPDOP</th><th>Minimum number of spacers</th></tr>
<tr><td>1b</td><td>The larger value of the following two values:
• Minimum number of spacers due to the D-PHY setting
• Minimum number of spacers due to the number of used lanes
Minimum number of spacers due to the D-PHY setting:
The minimum value can be found by using the ROUNDUP function that rounds up the decimal places.
ROUNDUP(((105/UI[ns] + 20) / 8), 0)</td></tr>
<tr><td></td><td>Minimum number of spacers due to the number of used lanes:
<table>
<tr><th>Number of used lanes</th><th>Minimum number of spacers</th></tr>
<tr><td>One lane</td><td>64</td></tr>
<tr><td>Two lanes</td><td>32</td></tr>
<tr><td>Four lanes</td><td>32</td></tr>
</table></td></tr>
</table></td></tr>
<tr><td>15</td><td>EPDOP</td><td>0b</td><td>RW</td><td>Selects an EPD option. Set this bit to 1b.
0b: D-PHY EPD Option 1 (0 cannot be set when EPDEN = 1)
1b: D-PHY EPD Option 2</td></tr>
<tr><td>14 to 0</td><td>SLP</td><td>All 0</td><td>RW</td><td>Specify the number of spacers inserted after a long packet.
This field is valid when EPDEN = 1.
When EPDEN = 1, the minimum number of spacers is the value described in the SSP field.
Setting a value less than the minimum value is prohibited.</td></tr>
</table>

35.1.6.8 Module Interrupt Status Register (CSI2nMIST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	VC3S	VC2S	VC1S	VC0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RXS	GSTS	PMS	—	—	—	—	DL3S	DL2S	DL1S	DL0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	VC3S	0b	R	When this bit is set to 1, there is an interrupt source regarding virtual channel 3. Check the CSI2nVCST3 register for the details of the interrupt source.
18	VC2S	0b	R	When this bit is set to 1, there is an interrupt source regarding virtual channel 2. Check the CSI2nVCST2 register for the details of the interrupt source.
17	VC1S	0b	R	When this bit is set to 1, there is an interrupt source regarding virtual channel 1. Check the CSI2nVCST1 register for the details of the interrupt source.
16	VC0S	0b	R	When this bit is set to 1, there is an interrupt source regarding virtual channel 0. Check the CSI2nVCST0 register for the details of the interrupt source.
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	RXS	0b	R	When this bit is set to 1, there is an interrupt source regarding the reception in general. Check the CSI2nRXST register for the details of the interrupt source.
9	GSTS	0b	R	When this bit is set to 1, there is an interrupt source regarding reception of generic short packets. Check the CSI2nGSST register for the details of the interrupt source.
8	PMS	0b	R	When this bit is set to 1, there is an interrupt source regarding the power management. Check the CSI2nPMST register for the details of the interrupt source.
7 to 4	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	DL3S	0b	R	When this bit is set to 1, there is an interrupt source regarding data lane 3. Check the CSI2nDLST3 register for the details of the interrupt source.
2	DL2S	0b	R	When this bit is set to 1, there is an interrupt source regarding data lane 2. Check the CSI2nDLST2 register for the details of the interrupt source.
1	DL1S	0b	R	When this bit is set to 1, there is an interrupt source regarding data lane 1. Check the CSI2nDLST1 register for the details of the interrupt source.
0	DL0S	0b	R	When this bit is set to 1, there is an interrupt source regarding data lane 0. Check the CSI2nDLST0 register for the details of the interrupt source.

35.1.6.9 Receive Data Type Enable Low Register (CSI2nDTEL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTEN[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTEN[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DTEN[31:0]	H'0000_000F	RW/R	Receive data of a packet whose data type is one of the following. 0b: Does not receive data. 1b: Receives data. *Note*: After multiple data types are selected here, only one data type is selected by the INF[5:0] bits in the ICnMC register in the image processing module.

Bit	Data Type	Attribute	Comment
31	YUV422 10-bit	RW	When this bit is set to 1b, data is output to the image processing module.
30	YUV422 8-bit	RW	
29	YUV420 10-bit (Chroma Shifted Pixel Sampling)	RW	
28	YUV420 8-bit (Chroma Shifted Pixel Sampling)	RW	
27	Reserved	RW	
26	Legacy YUV420 8-bit	RW	
25	YUV420 10-bit	RW	
24	YUV420 8-bit	RW	
23	Reserved	RW	
22	Generic Long Packet Data Type 4	RW	
21	Generic Long Packet Data Type 3	RW	
20	Generic Long Packet Data Type 2	RW	
19	Generic Long Packet Data Type 1	RW	
18-16	Reserved (Embedded 8-bit non Image Data, etc.)	RW	Fixed to 0 since this function is not support.
15	Generic Short Packet Code 8	RW	Fixed to 1 for each bits.
14	Generic Short Packet Code 7	RW	When each bits are set to 1b and CSI2nGSCT.GFIF = 1, data is output to the generic short packet FIFO.
13	Generic Short Packet Code 6	RW	
12	Generic Short Packet Code 5	RW	
11	Generic Short Packet Code 4	RW	
10	Generic Short Packet Code 3	RW	
9	Generic Short Packet Code 2	RW	
8	Generic Short Packet Code 1	RW	
7-4	Reserved	R	Fixed to 0.
3-0	Reserved	R	Fixed to 1.

35.1.6.10 Receive Data Type Enable High Register (CSI2nDTEH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTEN[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTEN[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DTEN[63:32]	H'0000_0000	RW	Receive data of a packet whose data type is one of the following. 0b: Does not receive data. 1b: Receives data.

Bit	Data Type	Attribute	Comment
31	Reserved	RW	Changing the reserved bit from the initial value is prohibited.
30-24	Reserved	RW	When this bit is set to 1b and CSI2nMCT0.RVMD = 1, data is output to the image processing module.
23	User Defined 8-bit Data Type 8	RW	When this bit is set to 1b, data is output to the image processing module.
22	User Defined 8-bit Data Type 7	RW	
21	User Defined 8-bit Data Type 6	RW	
20	User Defined 8-bit Data Type 5	RW	
19	User Defined 8-bit Data Type 4	RW	
18	User Defined 8-bit Data Type 3	RW	
17	User Defined 8-bit Data Type 2	RW	
16	User Defined 8-bit Data Type 1	RW	
15	RAW20	RW	
14	RAW16	RW	
13	RAW14	RW	
12	RAW12	RW	
11	RAW10	RW	
10	RAW8	RW	
9	RAW7	RW	
8	RAW6	RW	
7-5	Reserved	RW	
4	RGB888	RW	
3	RGB666	RW	
2	RGB565	RW	
1	RGB555	RW	
0	RGB444	RW	

35.1.6.11 Receive Status Register (CSI2nRXST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RACTDET	RACT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	FRM3	FRM2	FRM1	FRM0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 18	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	RACTDET	0b	R	When this bit is set to 1, CSI2nRXST.RACT = 1 has been detected. Write 1 to CSI2nRXSC.RACTDETC to clear this bit.
16	RACT	0b	R	Indicates that this IP is receiving packets.
15 to 4	—	H’000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	FRM3	0b	R	Indicates that frames on virtual channel = 3 are being processed.
2	FRM2	0b	R	Indicates that frames on virtual channel = 2 are being processed.
1	FRM1	0b	R	Indicates that frames on virtual channel = 1 are being processed.
0	FRM0	0b	R	Indicates that frames on virtual channel = 0 are being processed.

35.1.6.12 Receive Status Clear Register (CSI2nRXSC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RACTD ETC	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 18	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	RACTDETC	0b	R0W	Write 1 to this bit to clear CSI2nRXST.RACTDET. This bit cannot be modified to 0.
16 to 0	—	H‘0_0000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.1.6.13 Receive Interrupt Enable Register (CSI2nRXIE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RACTDETE	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 18	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	RACTDETE	0b	RW	Enables interrupts generated by CSI2nRXST.RACTDET. 0b: Does not assert csi2_int_rx when CSI2nRXST.RACTDET = 1. 1b: Asserts csi2_int_rx when CSI2nRXST.RACTDET = 1.
16 to 0	—	H’0_0000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.1.6.14 Data Lane (N) Status Register (CSI2nDLST(N))

(N) indicates a logical data lane number.

(N) = 0, 1, 2, or 3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	ULP(N)	—	—	—	—	—	—	RUL(N)	EUL(N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	EES(N)	ECT(N)	ESS(N)	ESH(N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	ULP(N)	0b	R	Indicates the RxUlpsEsc status of logical data lane (N).
23 to 20	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	RUL(N)	0b	R	When this bit is set to 1, logical data lane (N) has transitioned to the ULPS receive mode (RxUlpsEsc). Write 1 to CSI2nDLSC(N).RULC(N) to clear this bit.
16	EUL(N)	0b	R	When this bit is set to 1, logical data lane (N) has exited from the ULPS receive mode (RxUlpsEsc). Write 1 to CSI2nDLSC(N).EULC(N) to clear this bit.
15 to 12	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 4	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	EES(N)	0b	R	When this bit is set to 1, an escape mode entry error (ErrESC) has occurred on logical data lane (N). Write 1 to CSI2nDLSC(N).EESC(N) to clear this bit.

Bit	Bit Name	Initial Value	R/W	Description
2	ECT(N)	0b	R	When this bit is set to 1, a control error (ErrControl) has occurred on logical data lane (N). Write 1 to CSI2nDLSC(N).ECTC(N) to clear this bit.
1	ESS(N)	0b	R	When this bit is set to 1, an SoT synchronization error (ErrSotSyncHS) has occurred on logical data lane (N). Write 1 to CSI2nDLSC(N).ESSC(N) to clear this bit.
0	ESH(N)	0b	R	When this bit is set to 1, an SoT error (ErrSotHS) has occurred on logical data lane (N). Write 1 to CSI2nDLSC(N).ESHC(N) to clear this bit.

35.1.6.15 Data Lane (N) Status Clear Register (CSI2nDLSC(N))

(N) indicates a logical data lane number.

(N) = 0, 1, 2, or 3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RULC (N)	EULC (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	EESC (N)	ECTC (N)	ESSC (N)	ESHC (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	RULC(N)	0b	R0W	Write 1 to this bit to clear CSI2nDLST(N).RUL(N). This bit cannot be modified to 0.
16	EULC(N)	0b	R0W	Write 1 to this bit to clear CSI2nDLST(N).EUL(N). This bit cannot be modified to 0.
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 4	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	EESC(N)	0b	R0W	Write 1 to this bit to clear CSI2nDLST(N).EES(N). This bit cannot be modified to 0.
2	ECTC(N)	0b	R0W	Write 1 to this bit to clear CSI2nDLST(N).ECT(N). This bit cannot be modified to 0.
1	ESSC(N)	0b	R0W	Write 1 to this bit to clear CSI2nDLST(N).ESS(N). This bit cannot be modified to 0.
0	ESHC(N)	0b	R0W	Write 1 to this bit to clear CSI2nDLST(N).ESH(N). This bit cannot be modified to 0.

35.1.6.16 Data Lane (N) Interrupt Enable Register (CSI2nDLIE(N))

(N) indicates a logical data lane number.

(N) = 0, 1, 2, or 3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RULE (N)	EULE (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	EESE (N)	ECTE (N)	ESSE (N)	ESHE (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
18	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
17	RULE(N)	0b	RW	Enables interrupts generated by CSI2nDLST(N).RUL(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).RUL(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).RUL(N) = 1.
16	EULE(N)	0b	RW	Enables interrupts generated by CSI2nDLST(N).EUL(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).EUL(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).EUL(N) = 1.
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	—	H'0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 4	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	EESE(N)	0b	RW	Enables interrupts generated by CSI2nDLST(N).EES(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).EES(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).EES(N) = 1.
2	ECTE(N)	0b	RW	Enables interrupts generated by CSI2nDLST(N).ECT(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).ECT(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).ECT(N) = 1.
1	ESSE(N)	0b	RW	Enables interrupts generated by CSI2nDLST(N).ESS(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).ESS(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).ESS(N) = 1.
0	ESHE(N)	0b	RW	Enables interrupts generated by CSI2nDLST(N).ESH(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).ESH(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).ESH(N) = 1.

35.1.6.17 Virtual Channel (M) Status Register (CSI2nVCST(M))

(M) indicates a virtual channel number.

(M) = 0, 1, 2, or 3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	LER(M)	LSR(M)	FER(M)	FSR(M)	—	—	—	—	—	—	—	OVF(M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FRD(M)	FRS(M)	—	ECN(M)	ECC(M)	WCE(M)	IDE(M)	CRC(M)	ECD(M)	MLF(M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	LER(M)	0b	R	When this bit is set to 1, a packet whose data type is “Line End Code” has been received. Write 1 to CSI2nVCSC(M).LERC(M) to clear this bit.
26	LSR(M)	0b	R	When this bit is set to 1, a packet whose data type is “Line Start Code” has been received. Write 1 to CSI2nVCSC(M).LSRC(M) to clear this bit.
25	FER(M)	0b	R	When this bit is set to 1, a packet whose data type is “Frame End Code” has been received. Write 1 to CSI2nVCSC(M).FERC(M) to clear this bit.
24	FSR(M)	0b	R	When this bit is set to 1, a packet whose data type is “Frame Start Code” has been received. Write 1 to CSI2nVCSC(M).FSRC(M) to clear this bit.
23 to 17	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	OVF(M)	0b	R	When this bit is set to 1, the generic short packets on virtual channel M has been discarded due to an overflow of the generic short packet FIFO. Write 1 to CSI2nVCSC(M).OVFC(M) to clear this bit.
15 to 10	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	FRD(M)	0b	R	When this bit is set to 1, ErrFrameData has been detected. This bit is set to 1 if any of the following errors occurs between FS and FE: ● ErrCRC ● ErrWC ● ECC 2-bit error (when CSI2nMCT0.EDMD = 1) ● Reception of a packet of less than four bytes (when CSI2nMCT0.EDMD = 1) Note that this bit is set to 1 if Virtual Channel = M is between FS and FE because the virtual channel of a packet with an ECC 2-bit error or the virtual channel which received a packet of less than four bytes cannot be determined. Write 1 to CSI2nVCSC(M).FRDC(M) to clear this bit.

Bit	Bit Name	Initial Value	R/W	Description
8	FRS(M)	0b	R	When this bit is set to 1, ErrFrameSync has been detected. This bit is set to 1 if any of the following conditions occurs: ● After an FS is received, another FS has been received on the same virtual channel without receiving an FE. ● Although an FE is received, an FS has not been received on the same virtual channel. Write 1 to CSI2nVCSC(M).FRSC(M) to clear this bit.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	ECN(M)	0b	R	When this bit is set to 1, no error was found in the ECC check on the received packets. Write 1 to CSI2nVCSC(M).ECNC(M) to clear this bit.
5	ECC(M)	0b	R	When this bit is set to 1, a 1-bit error was found in the ECC check on the received packets and the error was corrected. Write 1 to CSI2nVCSC(M).ECCC(M) to clear this bit.
4	WCE(M)	0b	R	When this bit is set to 1, it was detected that the data payload length of the packet is shorter than the value indicated by WC. Write 1 to CSI2nVCSC(M).WCEC(M) to clear this bit.
3	IDE(M)	0b	R	When this bit is set to 1, an ErrID was detected and the packet was discarded. This bit is set to 1 if any of the following conditions occurs: ● A packet which is defined as "Reserved" in chapter 9, Low Level Protocol in the CSI-2 Specification has been received. – A packet whose Data Type is H’04 to H’07 has been received. – A packet whose Data Type is H’38 to H’3F has been received when CSI2nMCT0.RVMD = 0. ● A generic short packet (Data Type = H’08 to H’0F) for which the corresponding bit in CSI2nDTEL.DTEN is 0 has been received. ● A long packet (Data Type = H’10 to H’3F) for which the corresponding bit in CSI2nDTEL(H).DTEN is 0 has been received. Write 1 to CSI2nVCSC(M).IDEC(M) to clear this bit. Note that this bit may be set to 1 if more spacers than the count specified in CSI2nEPCT.SSP/SLP are received when using EPD option 2.
2	CRC(M)	0b	R	When this bit is set to 1, a CRC error was detected in the received packet. Write 1 to CSI2nVCSC(M).CRCC(M) to clear this bit.
1	ECD(M)	0b	R	When this bit is set to 1, a 2-bit error was found in the ECC check on the received packets. Because the virtual channel of the packet with an error cannot be determined, the ECD(M) field is set to 1 for all virtual channels. To clear the ECD(M) field for all virtual channels, write 1 to CSI2nVCSC(M).AECDC(M) for any virtual channel. To clear the ECD(M) field of virtual channel M, write 1 to CSI2nVCSC(M).ECDC(M).
0	MLF(M)	0b	R	When this bit is set to 1, a packet of less than 4 bytes has been received. Because the virtual channel of the packet with an error cannot be determined, the MLF(M) field is set to 1 for all virtual channels. To clear the MLF(M) field for all virtual channels, write 1 to CSI2nVCSC(M).AMLFC(M) for any virtual channel. To clear the MLF(M) field of virtual channel M, write 1 to CSI2nVCSC(M).MLFC(M).

35.1.6.18 Virtual Channel (M) Status Clear Register (CSI2nVCSC(M))

(M) indicates a virtual channel number.

(M) = 0, 1, 2, or 3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	LERC (M)	LSRC (M)	FERC (M)	FSRC (M)	—	—	—	—	—	—	—	OVFC (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R0W	R0W	R0W	R0W	R	R	R	R	R	R	R	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AECDC (M)	AMLFC (M)	—	—	—	—	FRDC (M)	FRSC (M)	—	ECNC (M)	ECCC (M)	WCEC (M)	IDEC (M)	CRCC (M)	ECDC (M)	MLFC (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W	R0W	R	R	R	R	R0W	R0W	R	R0W	R0W	R0W	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	LERC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).LER(M). This bit cannot be modified to 0.
26	LSRC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).LSR(M). This bit cannot be modified to 0.
25	FERC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).FER(M). This bit cannot be modified to 0.
24	FSRC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).FSR(M). This bit cannot be modified to 0.
23 to 17	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	OVFC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).OVF(M). This bit cannot be modified to 0.
15	AECDC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).ECD(M) for all virtual channels. This bit cannot be modified to 0.
14	AMLFC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).MLF(M) for all virtual channels. This bit cannot be modified to 0.
13 to 10	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	FRDC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).FRD(M). This bit cannot be modified to 0.
8	FRSC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).FRS(M). This bit cannot be modified to 0.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	ECNC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).ECN(M). This bit cannot be modified to 0.
5	ECCC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).ECC(M). This bit cannot be modified to 0.
4	WCEC(M)	0b	R0W	Write 1 to this bit to clear the CSI2nVCST(M).WCE(M). This bit cannot be modified to 0.

Bit	Bit Name	Initial Value	R/W	Description
3	IDEC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).IDE(M). This bit cannot be modified to 0.
2	CRCC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).CRC(M). This bit cannot be modified to 0.
1	ECDC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).ECD(M). This bit cannot be modified to 0.
0	MLFC(M)	0b	R0W	Write 1 to this bit to clear CSI2nVCST(M).MLF(M). This bit cannot be modified to 0.

35.1.6.19 Virtual Channel (M) Interrupt Enable Register (CSI2nVCIE(M))

(M) indicates a virtual channel number.

(M) = 0, 1, 2, or 3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	LERE (M)	LSRE (M)	FERE (M)	FSRE (M)	—	—	—	—	—	—	—	OVFE (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FRDE (M)	FRSE (M)	—	ECNE (M)	ECCE (M)	WCEE (M)	IDEE (M)	CRCE (M)	ECDE (M)	MLFE (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W	R	R	R	R	R	R	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27	LERE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).LER(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).LER(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).LER(M) = 1.
26	LSRE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).LSR(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).LSR(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).LSR(M) = 1.
25	FERE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).FER(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FER(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FER(M) = 1.
24	FSRE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).FSR(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FSR(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FSR(M) = 1.
23 to 17	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	OVFE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).OVF(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).OVF(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).OVF(M) = 1.
15 to 10	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	FRDE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).FRD(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FRD(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FRD(M) = 1.
8	FRSE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).FRS(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FRS(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FRS(M) = 1.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6	ECNE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).ECN(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).ECN(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).ECN(M) = 1.

Bit	Bit Name	Initial Value	R/W	Description
5	ECCE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).ECC(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).ECC(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).ECC(M) = 1.
4	WCEE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).WCE(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).WCE(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).WCE(M) = 1.
3	IDEE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).IDE(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).IDE(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).IDE(M) = 1.
2	CRCE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).CRC(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).CRC(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).CRC(M) = 1.
1	ECDE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).ECD(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).ECD(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).ECD(M) = 1.
0	MLFE(M)	0b	RW	Enables interrupts generated by CSI2nVCST(M).MLF(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).MLF(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).MLF(M) = 1.

35.1.6.20 Power Management Status Register (CSI2nPMST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	DLUL				—	—	—	—	DLSS			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLUL	CLSS	—	—	—	—	—	—	CUN	CUX	DUN	DUX	CSN	CSX	DSN	DSX
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 24	DLUL	H'0	R	Indicate the RxUlpsEsc status of a logical data lane. Bit 0: RxUlpsEsc of logical lane 0 Bit 1: RxUlpsEsc of logical lane 1 Bit 2: RxUlpsEsc of logical lane 2 Bit 3: RxUlpsEsc of logical lane 3
23 to 20	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19 to 16	DLSS	H'0	R	Indicate the stop state of a logical data lane. Bit 0: Stop state of logical lane 0 Bit 1: Stop state of logical lane 1 Bit 2: Stop state of logical lane 2 Bit 3: Stop state of logical lane 3
15	CLUL	0b	R	Indicates the inverted state of RxUlpsClkNot of the PPI clock lane. 0b: Not in ULPS 1b: In ULPS
14	CLSS	0b	R	Indicates the stop state of the PPI clock lane. 0b: Not in the stop state 1b: In the stop state
13 to 8	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CUN	0b	R	When this bit is set to 1, the clock lane has transitioned to ULPS mode (inversion of RxUlpsClkNot). Write 1 to CSI2nPMSC.CUNC to clear this bit.
6	CUX	0b	R	When this bit is set to 1, the mode of the clock lane has changed from ULPS mode (inversion of RxUlpsClkNot) to non-ULPS mode. Write 1 to CSI2nPMSC.CUXC to clear this bit.
5	DUN	0b	R	When this bit is set to 1, all valid data lanes have transitioned to ULPS mode. Write 1 to CSI2nPMSC.DUNC to clear this bit.
4	DUX	0b	R	When this bit is set to 1, the mode of all valid data lanes has changed from ULPS mode to a different state. Write 1 to CSI2nPMSC.DUXC to clear this bit.
3	CSN	0b	R	When this bit is set to 1, the clock lane has transitioned to the stop state. Write 1 to CSI2nPMSC.CSNC to clear this bit.
2	CSX	0b	R	When this bit is set to 1, the state of the clock lane has changed from the stop state to a state different from the stop state. Write 1 to CSI2nPMSC.CSXC to clear this bit.

Bit	Bit Name	Initial Value	R/W	Description
1	DSN	0b	R	When this bit is set to 1, all valid data lanes have transitioned to the stop state. Write 1 to CSI2nPMSC.DSNC to clear this bit.
0	DSX	0b	R	When this bit is set to 1, the mode of all valid data lanes has changed from the stop state to a different state. Write 1 to CSI2nPMSC.DSXC to clear this bit.

35.1.6.21 Power Management Status Clear Register (CSI2nPMSC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CUNC	CUXC	DUNC	DUXC	CSNC	CSXC	DSNC	DSXC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CUNC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.CUN. This bit cannot be modified to 0.
6	CUXC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.CUX. This bit cannot be modified to 0.
5	DUNC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.DUN. This bit cannot be modified to 0.
4	DUXC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.DUX. This bit cannot be modified to 0.
3	CSNC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.CSN. This bit cannot be modified to 0.
2	CSXC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.CSX. This bit cannot be modified to 0.
1	DSNC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.DSN. This bit cannot be modified to 0.
0	DSXC	0b	R0W	Write 1 to this bit to clear CSI2nPMST.DSX. This bit cannot be modified to 0.

35.1.6.22 Power Management Interrupt Enable Register (CSI2nPMIE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CUNE	CUXE	DUNE	DUXE	CSNE	CSXE	DSNE	DSXE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	CUNE	0b	RW	Enables interrupts generated by CSI2nPMST.CUN. 0b: Does not assert csi2_int_pm when CSI2nPMST.CUN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CUN = 1.
6	CUXE	0b	RW	Enables interrupts generated by CSI2nPMST.CUX. 0b: Does not assert csi2_int_pm when CSI2nPMST.CUX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CUX = 1.
5	DUNE	0b	RW	Enables interrupts generated by CSI2nPMST.DUN. 0b: Does not assert csi2_int_pm when CSI2nPMST.DUN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DUN = 1.
4	DUXE	0b	RW	Enables interrupts generated by CSI2nPMST.DUX. 0b: Does not assert csi2_int_pm when CSI2nPMST.DUX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DUX = 1.
3	CSNE	0b	RW	Enables interrupts generated by CSI2nPMST.CSN. 0b: Does not assert csi2_int_pm when CSI2nPMST.CSN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CSN = 1.
2	CSXE	0b	RW	Enables interrupts generated by CSI2nPMST.CSX. 0b: Does not assert csi2_int_pm when CSI2nPMST.CSX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CSX = 1.
1	DSNE	0b	RW	Enables interrupts generated by CSI2nPMST.DSN. 0b: Does not assert csi2_int_pm when CSI2nPMST.DSN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DSN = 1.
0	DSXE	0b	RW	Enables interrupts generated by CSI2nPMST.DSX. 0b: Does not assert csi2_int_pm when CSI2nPMST.DSX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DSX = 1.

35.1.6.23 Generic Short Packet Control Register (CSI2nGSCT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHTH						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
16	GFIF	1b	RW	When this bit is set to 1, the received generic short packet is stored in the generic short packet FIFO (GS FIFO).
15 to 7	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 0	SHTH	H'00	RW	Asserts CSI2nGSST.GTH while the number of packets stored in the generic short packet FIFO equals or exceeds the value of this field + 1. Set this field to 15 or less when using CSI2nGSST.GTH. Set the default value when not using CSI2nGSST.GTH.

35.1.6.24 Generic Short Packet Status Register (CSI2nGSST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STRDS	GCD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PNUM								—	—	—	GOV	—	—	GTH	GNE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
17	STRDS	0b	R	Indicates whether the received generic short packets can be stored in the generic short packet FIFO. 0b: Can be stored in the generic short packet FIFO. 1b: Cannot be stored in the generic short packet FIFO. This field indicates an inverted value of the store_en signal described in **Section 35.1.7.1** with a delay of internal latency. It may indicate the previous value depending on the signal change timing.
16	GCD	0b	R	When this bit is set to 1, the GS FIFO is being cleared by CSI2nGSCT.GFCLR.
15 to 8	PNUM	H’00	R	Indicate the number of packets stored in the generic short packet FIFO.
7 to 5	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	GOV	0b	R	When this bit is set to 1, the generic short packet FIFO has overflowed. Write 1 to CSI2nGSSC.GOVC to clear this bit.
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	GTH	0b	R	When this bit is set to 1, the number of packets stored in the generic short packet FIFO equals or exceeds “CSI2nGSCT.SHTH + 1”.
0	GNE	0b	R	When this bit is set to 1, the generic short packet FIFO is not empty.

35.1.6.25 Generic Short Packet Status Clear Register (CSI2nGSSC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GOVC	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	GOVC	0b	R0W	Write 1 to this bit to clear CSI2nGSST.GOV. This bit cannot be modified to 0.
3 to 0	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.1.6.26 Generic Short Packet Interrupt Enable Register (CSI2nGSIE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GOVE	—	—	GTHE	GNEE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	GOVE	0b	RW	Enables interrupts generated by CSI2nGSST.GOV. 0b: Does not assert csi2_int_gst when CSI2nGSST.GOV = 1. 1b: Asserts csi2_int_gst when CSI2nGSST.GOV = 1.
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	GTHE	0b	RW	Enables interrupts generated by CSI2nGSST.GTH. 0b: Does not assert csi2_int_gst when CSI2nGSST.GTH = 1. 1b: Asserts csi2_int_gst when CSI2nGSST.GTH = 1.
0	GNEE	0b	RW	Enables interrupts generated by CSI2nGSST.GNE. 0b: Does not assert csi2_int_gst when CSI2nGSST.GNE = 1. 1b: Asserts csi2_int_gst when CSI2nGSST.GNE = 1.

35.1.6.27 Generic Short Packet Register (CSI2nGSHT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SPVC				—	—	DTYP					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPDT															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 24	SPVC	H’0	R	Indicate the virtual channel.
23, 22	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 16	DTYP	H’08	R	Indicate the Data Type. H’08-H’0F
15 to 0	SPDT	H’0000	R	Indicate 16-bit user-defined data.

35.1.6.28 Generic Short Packet Information Update Register (CSI2nGSIU)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GFEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GFCLR	—	—	—	—	—	—	—	FINC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	GFEN	0b	R0W	Write 1 to this bit to re-enable storing generic short packets in the GS FIFO which has overflowed and been disabled. This bit cannot be modified to 0.
15 to 9	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	GFCLR	0b	RW	Clears the GS FIFO. Writing 1 to this bit requests the FIFO to be cleared. Writing 0 to this bit requests FIFO clearing to be canceled.
7 to 1	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FINC	0b	R0W	Write 1 to this bit to update the next packet with the data of the packet indicated in the CSI2nGSHT register. This bit cannot be modified to 0. Writing 1 to this bit while CSI2nGSST.PNUM = 0 is prohibited.

35.1.6.29 D-PHY Control Register 0 (CSIDPHYCTRL0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EN_LDO1200	EN_BGR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7 to 3	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	1b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	EN_LDO1200	0b	RW	Enables D-PHY power control 1. 0b: Disable 1b: Enable
0	EN_BGR	0b	RW	Enables D-PHY power control 0. 0b: Disable 1b: Enable

35.1.6.30 D-PHY Timing Register 0 (CSIDPHYTIM0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCLK_MISS								—	—	—	—	—	T_INIT		
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	T_INIT															
Initial Value	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TCLK_MISS	H’01	RW	Timing parameter*[1]
23 to 19	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18 to 0	T_INIT	H’0_2710	RW	Timing parameter*[1]

Note 1. The table below lists the recommended setting values. Please calculate and use linearly interpolated value for the setting of non-described in the below table between 80 Mbps and 360 Mbps.

Table 35.7 Recommended Setting Values of the D-PHY Timing Register

No.	Pin Name	Transmission Rate					Note
		80 Mbps	125 Mbps	250 Mbps	360 Mbps	Over 360 ~ 1500 Mbps	
1	T_INIT[18:0]	19’d79801 (H’137B9)	19’d79801 (H’137B9)	19’d79801 (H’137B9)	19’d79801 (H’137B9)	19’d79801 (H’137B9)	Min: 100 µs Recommend: 600 µs
2	TCLK_MISS[7:0]	8’d4 (H’4)	8’d4 (H’4)	8’d4 (H’4)	8’d4 (H’4)	8’d4 (H’4)	
3	TCLK_SETTLE[7:0]	8’d18 (H’12)	8’d18 (H’12)	8’d18 (H’12)	8’d18 (H’12)	8’d18 (H’12)	
4	THS_SETTLE[7:0]	8’d10 (H’A)	8’d11 (H’B)	8’d12 (H’C)	8’d12 (H’C)	8’d12 (H’C)	
5	TCLK_PREPARE[7:0]	8’d9 (H’9)	8’d10 (H’A)	8’d10 (H’A)	8’d10 (H’A)	8’d10 (H’A)	
6	THS_PREPARE[7:0]	8’d9 (H’9)	8’d9 (H’9)	8’d10 (H’A)	8’d10 (H’A)	8’d10 (H’A)	

35.1.6.31 D-PHY Timing Register 1 (CSIDPHYTIM1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	THS_PREPARE								TCLK_PREPARE							
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	THS_SETTLE								TCLK_SETTLE							
Initial Value	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	THS_PREPARE	H’04	RW	Timing parameter*[1]
23 to 16	TCLK_PREPARE	H’03	RW	Timing parameter*[1]
15 to 8	THS_SETTLE	H’09	RW	Timing parameter*[1]
7 to 0	TCLK_SETTLE	H’09	RW	Timing parameter*[1]

Note 1. **Table 35.7** lists the recommended setting values.

35.1.6.32 D-PHY Skew Adjustment Function (CSIDPHYSKW0)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	UTIL_DL3_SKW_ADJ			—	UTIL_DL2_SKW_ADJ			—	UTIL_DL1_SKW_ADJ			—	UTIL_DL0_SKW_ADJ		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18 to 16	—	000b	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14 to 12	UTIL_DL3_SKW_ADJ	000b	RW	Data lane 3 skew adjustment function Set the recommended value 001b. If required, change the value referring to **Figure 35.3**.
11	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10 to 8	UTIL_DL2_SKW_ADJ	000b	RW	Data lane 2 skew adjustment function Set the recommended value 001b. If required, change the value referring to **Figure 35.3**.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	UTIL_DL1_SKW_ADJ	000b	RW	Data lane 1 skew adjustment function Set the recommended value 001b. If required, change the value referring to **Figure 35.3**.
3	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	UTIL_DL0_SKW_ADJ	000b	RW	Data lane 0 skew adjustment function Set the recommended value 001b. If required, change the value referring to **Figure 35.3**.

The skew adjustment function adjusts the skew by using the circuit shown in **Figure 35.3**.

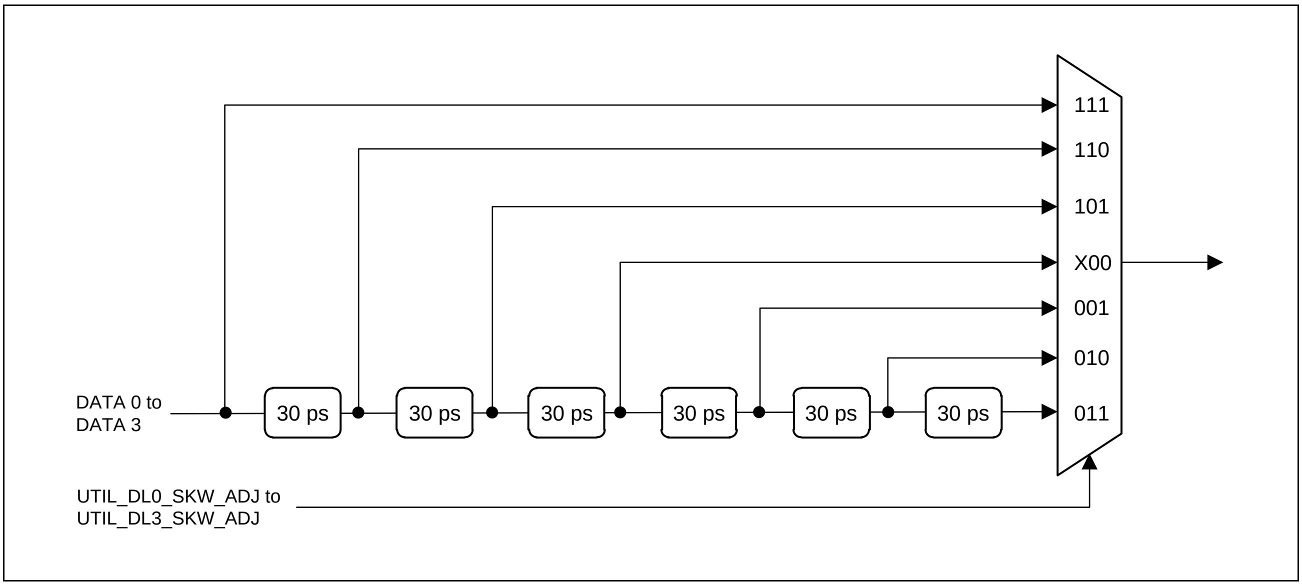


Figure 35.3 Skew Adjustment Circuit

35.1.7 Operation

35.1.7.1 Data Reception

(1) Common Between Short Packets and Long Packets

Figure 35.4 shows the flow when receiving a packet whose data type is A and virtual channel is M.

This flow is common between short packets and long packets.

Upon receiving a packet, the CRU checks the length of the received packet.

If the length of the received packet is less than the packet header length (4 bytes), a virtual channel cannot be determined, resulting in setting CSI2nVCST(M).MLF(M) to 1 for all virtual channels. In addition, this packet is discarded because it is invalid.

When the length of the received packet is 4 bytes or more, ECC checking is performed. The check results are indicated in the CSI2nVCST(M) register.

- 2-bit error: Because the virtual channel of the packet with an error cannot be determined, CSI2nVCST(M).ECD(M) is set to 1 for all virtual channels.
  In addition, the packet is discarded because error correction cannot be performed.
- 1-bit error: CSI2nVCST(M).ECC(M) is set to 1 for the relevant virtual channel.
  Error correction is performed because it is possible.
  No error: CSI2nVCST(M).ECN(M) is set to 1 for the relevant virtual channel.

If the ECC check results are “1-bit error” or “no error”, the Data Type field of the packet header is analyzed.

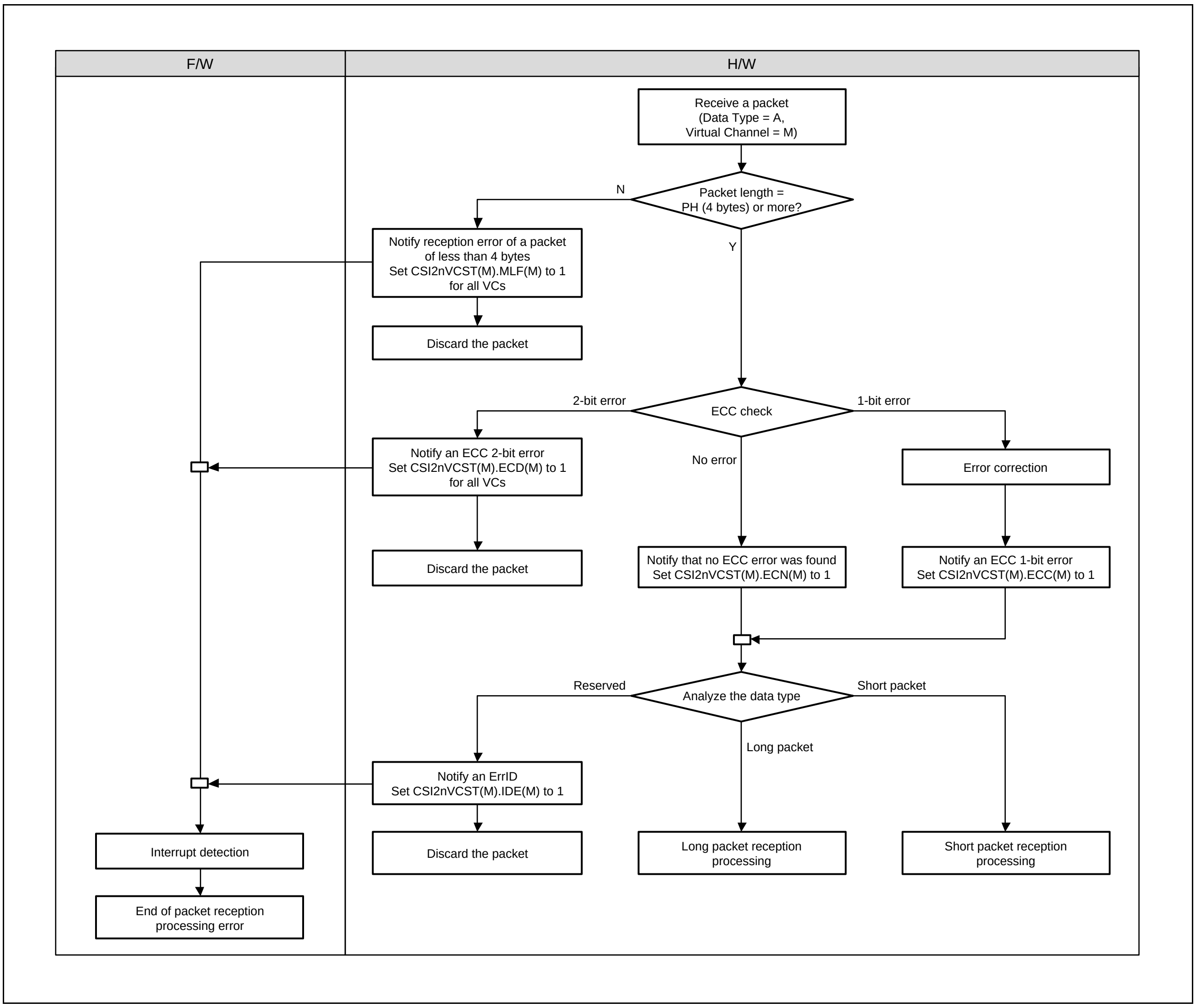


Figure 35.4 Packet Reception Processing

(2) Generic Short Packet Data Type Code

Figure 35.5 shows the flow when receiving a packet whose data type is generic short packet code 1 to 8 and virtual channel is M.

When the corresponding bit in CSI2nDTEL.DTEN is set to 0, CSI2nVCST(M).IDE(M)) is asserted and the packet is discarded.

When the corresponding bit in CSI2nDTEL.DTEN is set to 1 and CSI2nGSCT.GFIF is set to 1, the generic short packet is output to the generic short packet FIFO. For details on the operation when using the generic short packet FIFO, see the flow in **Figure 35.5**.

The initial value of internal signal store_en which is described in the flow is 1. When store_en is set to 1, packets can be stored in the generic short packet FIFO. When it is set to 0, packets cannot be stored in the generic short packet FIFO.

- store_en is set to 0 when the generic short packet FIFO has overflowed.
- store_en is set to 1 when 1 is written to CSI2nGSIU.GFEN.

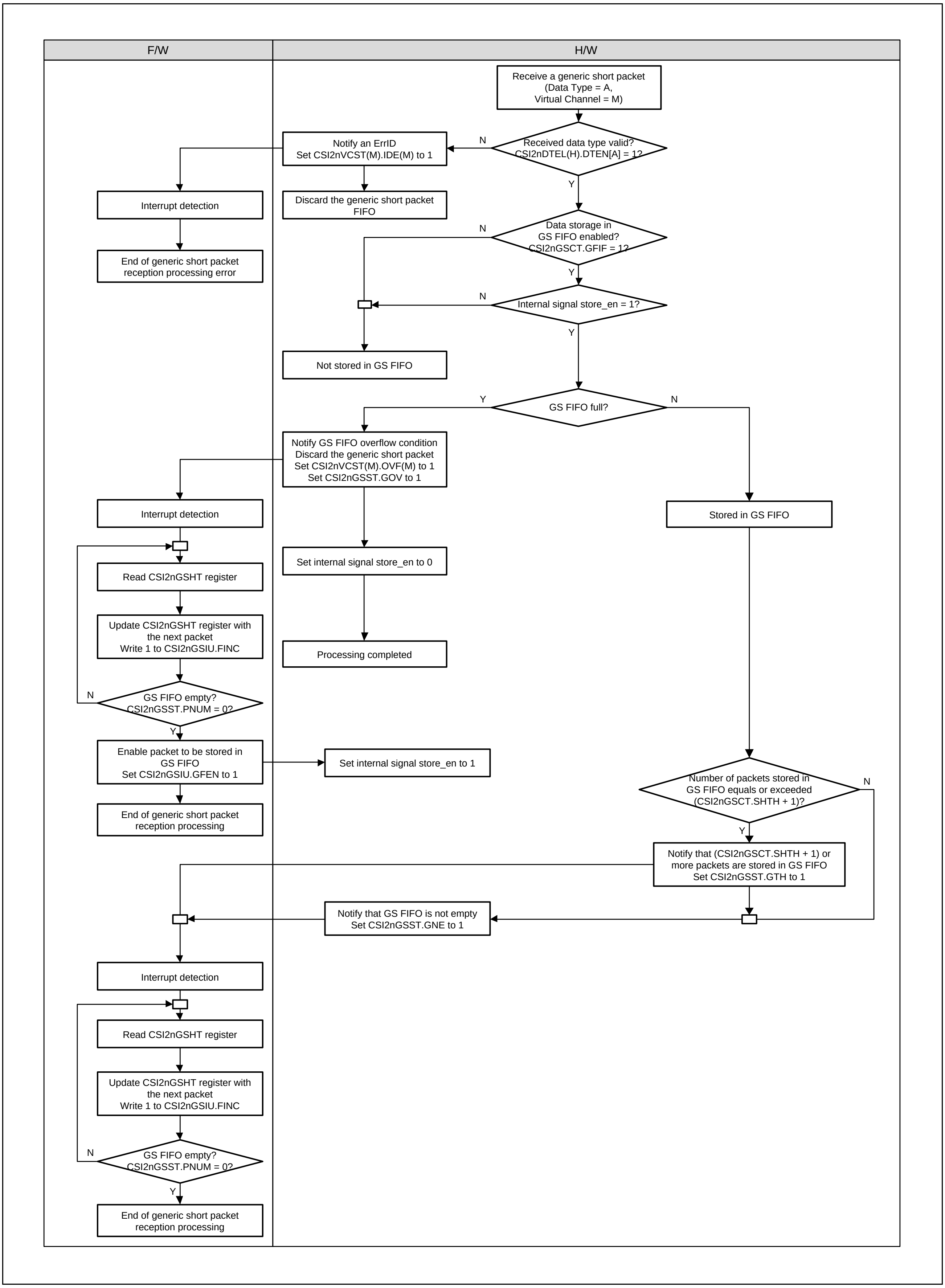


Figure 35.5 Generic Short Packet Receive Processing

(3) Clearing the Generic Short Packet FIFO

Figure 35.6 shows how the generic short packet FIFO is cleared.

Writing 1 to CSI2nGSIU.GFCLR starts clearing the generic short packet FIFO.

After requesting to clear the generic short packet FIFO, check if CSI2nGSST.GCD is set to 1 to make sure that the FIFO is cleared. Then, writing 0 to CSI2nGSIU.GFCLR cancels the request for clearing the generic short packet FIFO. Check if CSI2nGSST.GCD is set to 0 to make sure that the request for clearing the generic short packet FIFO is canceled.

When CSI2nMCG.GSNM is set to 0d, it is prohibited to clear the generic short packet FIFO.

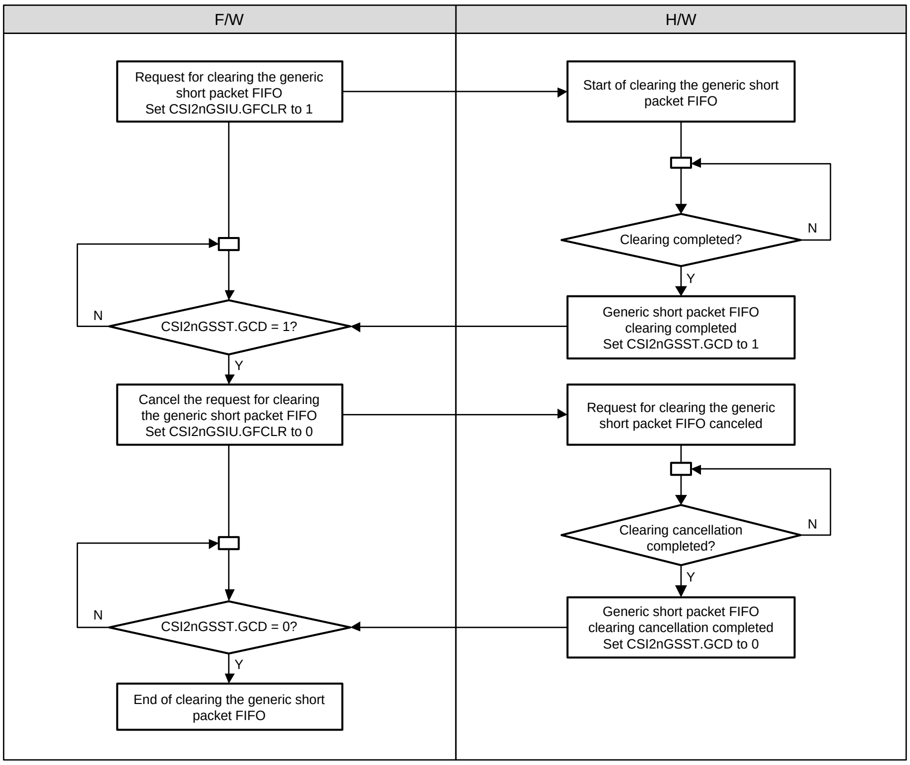


Figure 35.6 Clearing the Generic Short Packet FIFO

35.1.8 Interrupt

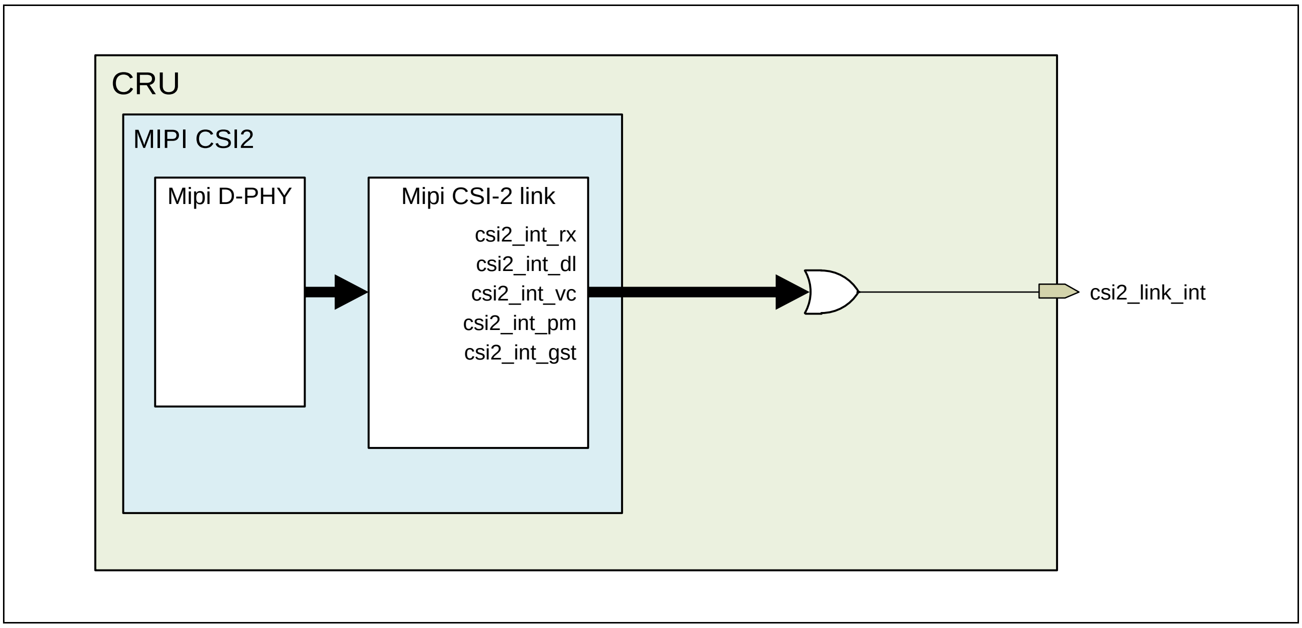


Figure 35.7 Interrupt Structure

Table 35.8 Interruption Status and Generation Condition (1/3)

Interrupt Status	Interrupt Source Register Field	Interrupt Enable Register Field	Generation Condition
CSI2nMIST.RXS	CSI2nRXST.RACTDET	CSI2nRXIE.RACTDETE	This IP received a packet.
CSI2nMIST.DLS0	CSI2nDLST0.ESH0	CSI2nDLIE0.ESHE0	ErrSotHS occurred on DL0.
	CSI2nDLST0.ESS0	CSI2nDLIE0.ESSE0	ErrSotSyncHS occurred on DL0.
	CSI2nDLST0.ECT0	CSI2nDLIE0.ECTE0	ErrControl occurred on DL0.
	CSI2nDLST0.EES0	CSI2nDLIE0.EESE0	ErrESC occurred on DL0.
	CSI2nDLST0.EUL0	CSI2nDLIE0.EULE0	DL0 exited from the ULPS.
	CSI2nDLST0.RUL0	CSI2nDLIE0.RULE0	DL0 transitioned to the ULPS.
CSI2nMIST.DLS1	CSI2nDLST1.ESH1	CSI2nDLIE1.ESHE1	ErrSotHS occurred on DL1.
	CSI2nDLST1.ESS1	CSI2nDLIE1.ESSE1	ErrSotSyncHS occurred on DL1.
	CSI2nDLST1.ECT1	CSI2nDLIE1.ECTE1	ErrControl occurred on DL1.
	CSI2nDLST1.EES1	CSI2nDLIE1.EESE1	ErrESC occurred on DL1.
	CSI2nDLST1.EUL1	CSI2nDLIE1.EULE1	DL1 exited from the ULPS.
	CSI2nDLST1.RUL1	CSI2nDLIE1.RULE1	DL1 transitioned to the ULPS.
…	…	…	…
CSI2nMIST.DLS3	CSI2nDLST3.ESH3	CSI2nDLIE3.ESHE3	ErrSotHS occurred on DL3.
	CSI2nDLST3.ESS3	CSI2nDLIE3.ESSE3	ErrSotSyncHS occurred on DL3.
	CSI2nDLST3.ECT3	CSI2nDLIE3.ECTE3	ErrControl occurred on DL3.
	CSI2nDLST3.EES3	CSI2nDLIE3.EESE3	ErrESC occurred on DL3.
	CSI2nDLST3.EUL3	CSI2nDLIE3.EULE3	DL3 exited from the ULPS.
	CSI2nDLST3.RUL3	CSI2nDLIE3.RULE3	DL3 transitioned to the ULPS.

Table 35.8 Interruption Status and Generation Condition (2/3)

Interrupt Status	Interrupt Source Register Field	Interrupt Enable Register Field	Generation Condition
CSI2nMIST.VCS0	CSI2nVCST0.MLF0	CSI2nVCIE0.MLFE0	Malformed packet was received on VC0.
	CSI2nVCST0.ECD0	CSI2nVCIE0.ECDE0	ErrEccDouble occurred on VC0.
	CSI2nVCST0.CRC0	CSI2nVCIE0.CRCE0	ErrCrc occurred on VC0.
	CSI2nVCST0.IDE0	CSI2nVCIE0.IDEE0	ErrID occurred on VC0.
	CSI2nVCST0.WCE0	CSI2nVCIE0.WCEE0	ErrWC occurred on VC0.
	CSI2nVCST0.ECC0	CSI2nVCIE0.ECCE0	ErrEccCorrected occurred on VC0.
	CSI2nVCST0.ECN0	CSI2nVCIE0.ECNE0	ErrEccNoError occurred on VC0.
	CSI2nVCST0.FRS0	CSI2nVCIE0.FRSE0	ErrFrameSync occurred on VC0.
	CSI2nVCST0.FRD0	CSI2nVCIE0.FRDE0	ErrFrameData occurred on VC0.
	CSI2nVCST0.OVF0	CSI2nVCIE0.OVFE0	Generic short packets on VC0 were discarded because GS FIFO overflowed.
	CSI2nVCST0.FSR0	CSI2nVCIE0.FSRE0	Frame Start was received on VC0.
	CSI2nVCST0.FER0	CSI2nVCIE0.FERE0	Frame End was received on VC0.
	CSI2nVCST0.LSR0	CSI2nVCIE0.LSRE0	Line Start was received on VC0.
	CSI2nVCST0.LER0	CSI2nVCIE0.LERE0	Line End was received on VC0.
CSI2nMIST.VCS1	CSI2nVCST1.MLF1	CSI2nVCIE1.MLFE1	Malformed packet was received on VC1.
	CSI2nVCST1.ECD1	CSI2nVCIE1.ECDE1	ErrEccDouble occurred on VC1.
	CSI2nVCST1.CRC1	CSI2nVCIE1.CRCE1	ErrCrc occurred on VC1.
	CSI2nVCST1.IDE1	CSI2nVCIE1.IDEE1	ErrID occurred on VC1.
	CSI2nVCST1.WCE1	CSI2nVCIE1.WCEE1	ErrWC occurred on VC1.
	CSI2nVCST1.ECC1	CSI2nVCIE1.ECCE1	ErrEccCorrected occurred on VC1.
	CSI2nVCST1.ECN1	CSI2nVCIE1.ECNE1	ErrEccNoError occurred on VC1.
	CSI2nVCST1.FRS1	CSI2nVCIE1.FRSE1	ErrFrameSync occurred on VC1.
	CSI2nVCST1.FRD1	CSI2nVCIE1.FRDE1	ErrFrameData occurred on VC1.
	CSI2nVCST1.OVF1	CSI2nVCIE1.OVFE1	Generic short packets on VC1 were discarded because GS FIFO overflowed.
	CSI2nVCST1.FSR1	CSI2nVCIE1.FSRE1	Frame Start was received on VC1.
	CSI2nVCST1.FER1	CSI2nVCIE1.FERE1	Frame End was received on VC1.
	CSI2nVCST1.LSR1	CSI2nVCIE1.LSRE1	Line Start was received on VC1.
	CSI2nVCST1.LER1	CSI2nVCIE1.LERE1	Line End was received on VC1.
…	…	…	…

Table 35.8 Interruption Status and Generation Condition (3/3)

Interrupt Status	Interrupt Source Register Field	Interrupt Enable Register Field	Generation Condition
CSI2nMIST.VCS3	CSI2nVCST3.MLF3	CSI2nVCIE3.MLFE3	Malformed packet was received on VC3.
	CSI2nVCST3ECD3	CSI2nVCIE3.ECDE3	ErrEccDouble occurred on VC3.
	CSI2nVCST3.CRC3	CSI2nVCIE3.CRCE3	ErrCrc occurred on VC3.
	CSI2nVCST3.IDE3	CSI2nVCIE3.IDEE3	ErrID occurred on VC3.
	CSI2nVCST34.WCE3	CSI2nVCIE3.WCEE3	ErrWC occurred on VC3.
	CSI2nVCST3.ECC3	CSI2nVCIE3.ECCE3	ErrEccCorrected occurred on VC3.
	CSI2nVCST3.ECN3	CSI2nVCIE3.ECNE3	ErrEccNoError occurred on VC3.
	CSI2nVCST3.FRS3	CSI2nVCIE3.FRSE3	ErrFrameSync occurred on VC3.
	CSI2nVCST3.FRD3	CSI2nVCIE3.FRDE3	ErrFrameData occurred on VC3.
	CSI2nVCST3OVF3	CSI2nVCIE3.OVFE3	Generic short packets on VC3 were discarded because GS FIFO overflowed.
	CSI2nVCST3.FSR3	CSI2nVCIE3.FSRE3	Frame Start was received on VC3.
	CSI2nVCST3.FER3	CSI2nVCIE3.FERE3	Frame End was received on VC3.
	CSI2nVCST3.LSR3	CSI2nVCIE3.LSRE3	Line Start was received on VC3.
	CSI2nVCST3.LER3	CSI2nVCIE3.LERE3	Line End was received on VC3.
CSI2nMIST.PMS	CSI2nPMST.DSX	CSI2nPMIE.DSXE	All data lanes exited from the stop state.
	CSI2nPMST.DSN	CSI2nPMIE.DSNE	All data lanes transitioned to the stop state.
	CSI2nPMST.CSX	CSI2nPMIE.CSXE	The clock lane exited from the stop state.
	CSI2nPMST.CSN	CSI2nPMIE.CSNE	The clock lane transitioned to the stop state.
	CSI2nPMST.DUX	CSI2nPMIE.DUXE	All data lanes exited from the ULPS.
	CSI2nPMST.DUN	CSI2nPMIE.DUNE	All data lanes transitioned to the ULPS.
	CSI2nPMST.CUX	CSI2nPMIE.CUXE	The clock lane exited from the ULPS.
	CSI2nPMST.CUN	CSI2nPMIE.CUNE	The clock lane transitioned to the ULPS.
CSI2nMIST.GSTS	CSI2nGSST.GNE	CSI2nGSIE.GNEE	GS FIFO is not empty.
	CSI2nGSST.GTH	CSI2nGSIE.GTHE	Data stored in GS FIFO equals or exceeds the threshold.
	CSI2nGSST.GOV	CSI2nGSIE.GOVE	GS FIFO overflowed.

35.2 Image Processing

35.2.1 Overview

The image processing module is a data conversion module equipped with pixel color space conversion, LUT, pixel format conversion, image clipping, frame subsampling, demosaicing, linear matrix, statistics, etc. An MIPI CSI-2 (**Section 35.1**) input and parallel (including ITU-R BT.656) input are provided as the image sensor interface. Either of the two inputs is selected and stored in the internal FIFO (RAM), and then the data is transferred to the external memory by using the internal AXI bus. The MIPI CSI-2 input processes only a single data type on one channel from virtual channels CH0 to CH3. Some parameters allow changes on a per-frame basis. If multiple virtual channels (with the same image size) are to be processed, channels are switched at the frame boundary by changing register settings.

35.2.1.1 Features

- The operating ranges are as follows:
  - Maximum number of valid pixels: QVGA (320 × 240) to 5 Mpixels
  - Range of valid pixels in the horizontal direction: 320 to 2800 pixels
  - Range of valid pixels in the vertical direction: 240 to 4095 lines
  - The maximum transfer rate depends on the input format, see **Section 35.1.2** for details.
  - Parallel (including ITU-R BT.656) input: 13.5 MHz to 74.25 MHz
- Input data format (type): See **Table 35.12**.
  - MIPI CSI-2 input: YUV, RGB, RAW, etc.
  - Parallel input: YCbCr, 16-bit binary, etc.
- Output data format (type): Only little endian is supported.
  - YCbCr422/YCbCr420, 8-bit multiplexed (Y, U/V order swapping supported)
  - YC separation from YCbCr422 (separated into Y and CbCr components)
  - Interlaced (parallel input only) YCbCr420 output without YC separation is supported.
  - Y component extraction from YCbCr422
  - U/V of the YUV422/YCbCr420 output is expressed in offset binary or two’s complement notation.
  - RGB-888 (24 bits/pixel)
  - RGB-888 (32 bits/pixel)
  - ARGB-8888 (32 bits/pixel)
  - RAW8,10,12,14,16
  - 16-bit binary (parallel input only)
  - MIPI CSI-2 Recommended Memory Storage Format
- Color space conversion
  - YCbCr422 to RGB888
  - RGB888 to YCbCr422
- Image format conversion
  - YCbCr422 → YC separation (separated into Y and CbCr components)

  - YCbCr422 → Y component extraction
  - YCbCr422 → YCbCr420
    Interlaced (parallel input only) YCbCr420 output without YC separation is supported.
  - RGB-888 → RGB-888 (32 bits/pixel)
  - RGB-888 → ARGB-8888 (32 bits/pixel): B First A Last format
  - RGB-888 → ARGB-8888 (32 bits/pixel): A First B Last format
  - RGB-565 → RGB-888 conversion (intermediate processing)
  - RGB-666 → RGB-888 conversion (intermediate processing)
- Conversion 10-bit to 8-bit by using a lookup table (LUT)
- Data clipping (See **Section 35.2.4.13**)
- Image clipping
- Frame subsampling
- Bypassing each function (function bypassed)
- Pattern generator (for debugging)
  - Output Y, U, V register values.
- Demosaicing
  - RAW8, 10, 12, 14, and 16 to RGB888
- Statistics (statistical processing operations: horizontal detection area: 320 to 2048 pixels; vertical detection area: 240 to 4080)
  - Accumulation in the (16×16, 32×32, 64×64, and 128×128) areas for each of R, G, and B from RAW8, 10, 12, 14, and 16 data
  - Summation of absolute values for differences from adjacent pixel values of G from RAW8, 10, 12, 14, and 16 data
- Linear Matrix (Color correction by linear matrix operations)
  - Offset calculation included
- FIFO:
  - For image data: 8 K bytes
  - For statistics data: 512 bytes
- AXI master: Two channels (image processing, statistics system)
- APB slave: Only 32-bit access is supported.

35.2.1.2 Block Diagram

Below figure shows a block diagram of the image processing module.

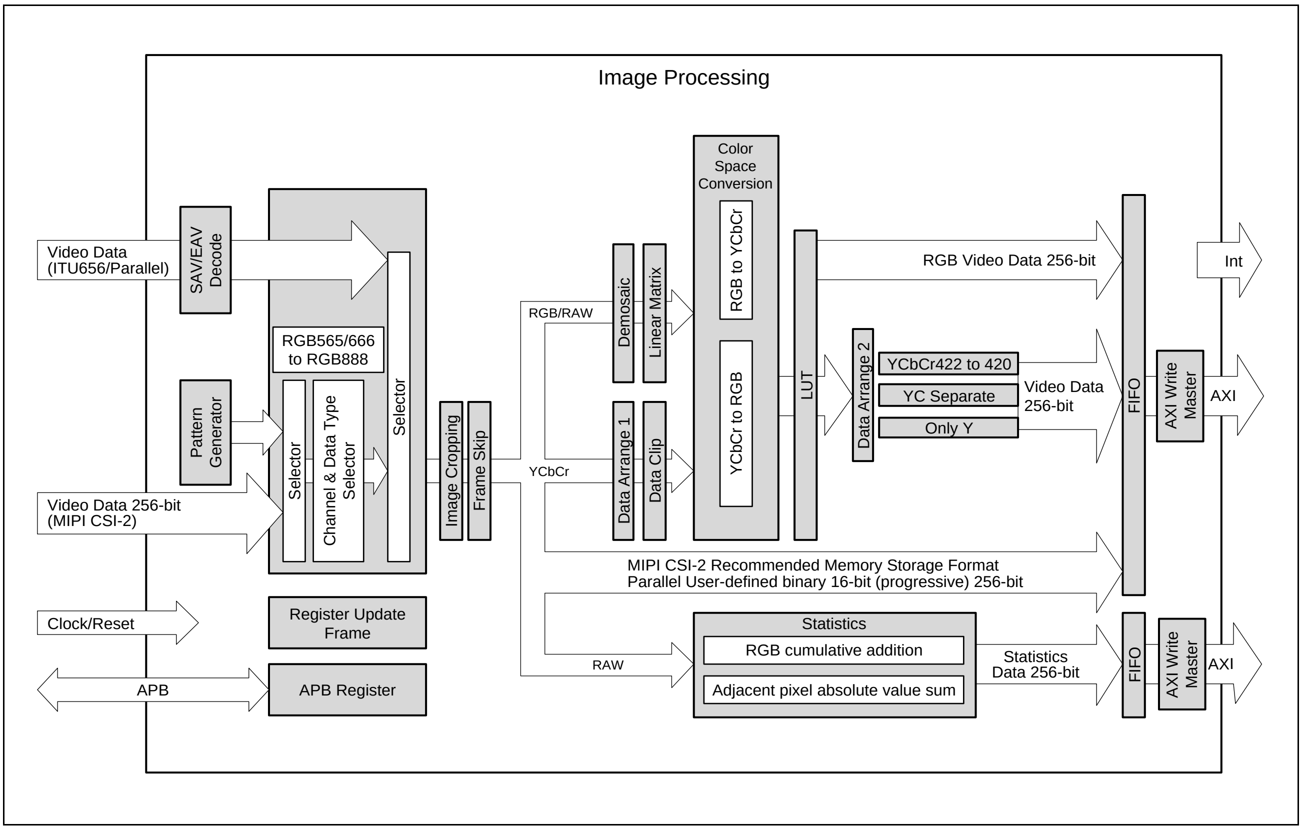


Figure 35.8 Block Diagram of the Image Processing Module

35.2.1.3 Parallel Inputs

Table 35.9 lists the input and output pins used by the image processing module. Because these pins are multiplexed with other pins, they need settings. The MIPI CSI-2 signal is input to the image processing module through the MIPI CSI-2 receiver module.

Table 35.9 Parallel Input Pins

Classification	Pin Name	In/Out	Sync Clock	Sync Edge	Description
Parallel Video Interface	CAM_PCLK	In	—	—	External Parallel Video Clock
	CAM_VSYNC	In	CAM_PCLK	pos	Vsync External Video Input
	CAM_FIELD	In	CAM_PCLK	pos	Field External Video Input
	CAM_HREF	In	CAM_PCLK	pos	Data Enable & Hsync External Video Input
	CAM_DATA15 to CAM_DATA0	In	CAM_PCLK	pos	Data External Video Input When using ITU-R BT.656. Use CAM_DATA9 to CAM_DATA0 (10 bits) Use CAM_DATA9 to CAM_DATA2 (8 bits)

(1) Parallel Data Input Timing

Figure 35.9 and **Figure 35.10** show the input data timing except for the ITU-R BT.656 input.

Figure 35.9 is an example when ICnPIFC.PINF[3:0] = 0010b, ICnPIFC.ITL[2:0] = 101b, ICnPIFC.VSPOL = 0 and ICnPIFC.ENPOL = 0.

Figure 35.10 is an example when ICnPIFC.PINF[3:0] = 0010b, ICnPIFC.ITL[2:0] = 000b, ICnPIFC.VSPOL = 0 and ICnPIFC.ENPOL = 0.

The polarity of CAM_VSYNC and CAM_HREF can be inverted by register settings.

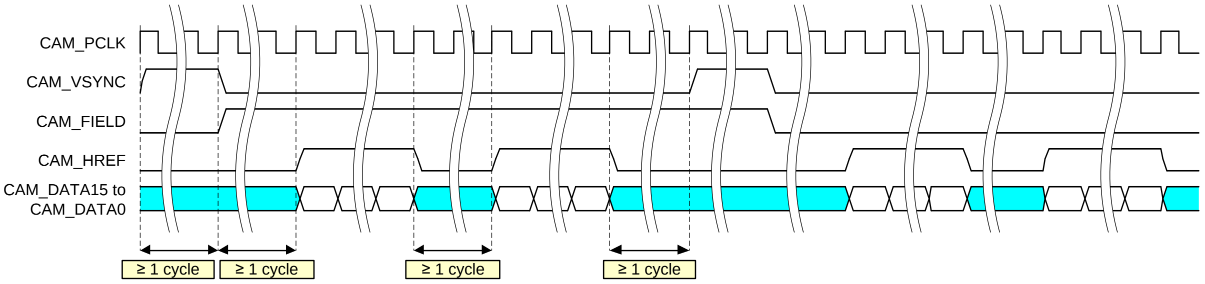


Figure 35.9 Parallel Data Input Timing (Interlaced)

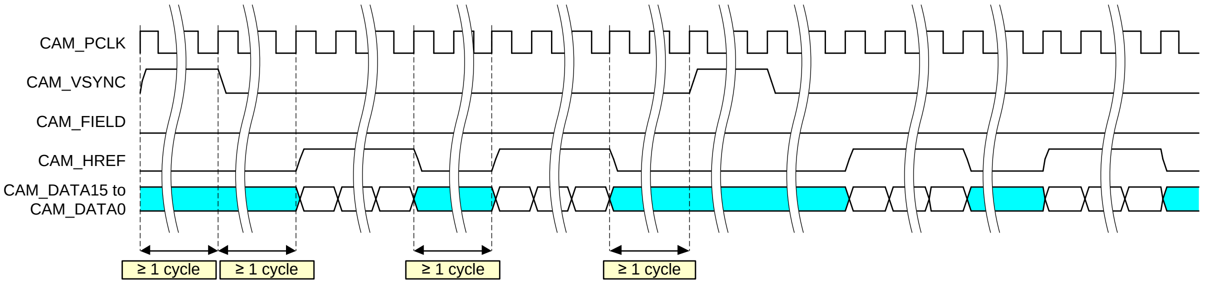


Figure 35.10 Parallel Data Input Timing (Progressive)

(2) Parallel Input Format Correspondence Table

Table 35.10 is a parallel input format correspondence table.

Table 35.10 Parallel Input Format Correspondence Table

Case		CAM_DATA15 to CAM_DATA0															
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
YCbCr separation, 8 bits	Cb 1st	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
		Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
	Cr 1st	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
		Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
YCbCr multiplexing, 8 bits	Cb 1st	—	—	—	—	—	—	—	—	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
		—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	—	—	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
		—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
	Cr 1st	—	—	—	—	—	—	—	—	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
		—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	—	—	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
		—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
	Cb 2nd	—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	—	—	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
		—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	—	—	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
	Cr 2nd	—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	—	—	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
		—	—	—	—	—	—	—	—	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	—	—	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
YCbCr multiplexing, 10 bits	Cb 1st	—	—	—	—	—	—	Cb9	Cb8	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
		—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	Cr9	Cr8	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
		—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
	Cr 1st	—	—	—	—	—	—	Cr9	Cr8	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
		—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	Cb9	Cb8	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
		—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
	Cb 2nd	—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	Cb9	Cb8	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
		—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	Cr9	Cr8	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
	Cr 2nd	—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	Cr9	Cr8	Cr7	Cr6	Cr5	Cr4	Cr3	Cr2	Cr1	Cr0
		—	—	—	—	—	—	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
		—	—	—	—	—	—	Cb9	Cb8	Cb7	Cb6	Cb5	Cb4	Cb3	Cb2	Cb1	Cb0
ITU-R BT.656*[1]	10 bits	—	—	—	—	—	—	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	8 bits	—	—	—	—	—	—	D7	D6	D5	D4	D3	D2	D1	D0	—	—
Binary, 16 bits*[2]		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Note 1. Signals D9 to D0 are connected according to the number of Data bit number in TABLE 2: “Video timing reference codes” of ITU-R BT.656-5 standard.

Note 2. The 16-bit binary data is packed with 1-pixel data, such as user-defined RGB, YCbCr, etc., in the 16 bits.

35.2.2 Register Configuration

The following shows the base address of this module.

Module Name	Address
Image Processing	H'0_1083_0000 (Cortex-A55 Address Space) H'4083_0000 (Cortex-M33 Address Space Non-Secure) H'5083_0000 (Cortex-M33 Address Space Secure)

The register values at addresses H'1083_0208 to H'1083_0244 can be changed through frame synchronization. For the changing flows, see **Section 35.3.3.1** and **Section 35.3.3.2**.

Table 35.11 Register Configuration of the Image Processing Module (1/4)

Register Name	Register Symbol Name	Address	R/W	Initial Value
Common				
CRU Control Register	CRUnCTRL	H'0000	R/W	H'0000_0000
CRU Interrupt Enable Register	CRUnIE	H'0004	R/W	H'0000_0000
CRU Interrupt Status Register	CRUnINTS	H'0008	R/W	H'0000_0000
CRU Reset Register	CRUnRST	H'000C	R/W	H'0000_0000
Reserved		H'000C-H'007F	—	—
CRU General Read/Write Register	CRUnCOM	H'0080	R/W	H'0000_0000
Reserved		H'0084-H'00FF	—	—
AXI Master				
Memory Bank 1 Base Address (Lower) Register for CRU Image Data	AMnMB1ADDRL	H'0100	R/W	H'0000_0000
Memory Bank 1 Base Address (Higher) Register for CRU Image Data	AMnMB1ADDRH	H'0104	R/W	H'0000_0000
Memory Bank 2 Base Address (Lower) Register for CRU Image Data	AMnMB2ADDRL	H'0108	R/W	H'0000_0000
Memory Bank 2 Base Address (Higher) Register for CRU Image Data	AMnMB2ADDRH	H'010C	R/W	H'0000_0000
Memory Bank 3 Base Address (Lower) Register for CRU Image Data	AMnMB3ADDRL	H'0110	R/W	H'0000_0000
Memory Bank 3 Base Address (Higher) Register for CRU Image Data	AMnMB3ADDRH	H'0114	R/W	H'0000_0000
Memory Bank 4 Base Address (Lower) Register for CRU Image Data	AMnMB4ADDRL	H'0118	R/W	H'0000_0000
Memory Bank 4 Base Address (Higher) Register for CRU Image Data	AMnMB4ADDRH	H'011C	R/W	H'0000_0000
Memory Bank 5 Base Address (Lower) Register for CRU Image Data	AMnMB5ADDRL	H'0120	R/W	H'0000_0000
Memory Bank 5 Base Address (Higher) Register for CRU Image Data	AMnMB5ADDRH	H'0124	R/W	H'0000_0000
Memory Bank 6 Base Address (Lower) Register for CRU Image Data	AMnMB6ADDRL	H'0128	R/W	H'0000_0000
Memory Bank 6 Base Address (Higher) Register for CRU Image Data	AMnMB6ADDRH	H'012C	R/W	H'0000_0000
Memory Bank 7 Base Address (Lower) Register for CRU Image Data	AMnMB7ADDRL	H'0130	R/W	H'0000_0000
Memory Bank 7 Base Address (Higher) Register for CRU Image Data	AMnMB7ADDRH	H'0134	R/W	H'0000_0000

Table 35.11 Register Configuration of the Image Processing Module (2/4)

Register Name	Register Symbol Name	Address	R/W	Initial Value
Memory Bank 8 Base Address (Lower) Register for CRU Image Data	AMnMB8ADDRL	H’0138	R/W	H’0000_0000
Memory Bank 8 Base Address (Higher) Register for CRU Image Data	AMnMB8ADDRH	H’013C	R/W	H’0000_0000
UV Data Address Offset (Lower) Register for CRU Image Data	AMnUVAOFL	H’0140	R/W	H’0000_0000
UV Data Address Offset (Higher) Register for CRU Image Data	AMnUVAOFH	H’0144	R/W	H’0000_0000
Memory Bank Enable Register for CRU Image Data	AMnMBVALID	H’0148	R/W	H’0000_0001
Memory Bank Status Register for CRU Image Data	AMnMBS	H’014C	R	H’0000_0000
AXI Master Transfer Setting Register for CRU Image Data	AMnAXIATTR	H’0158	R/W	H’0000_0150
Reserved		H’015C-H’015F	—	—
AXI Master FIFO Setting Register for CRU Image Data	AMnFIFO	H’0160	R/W	H’0000_0001
AXI Master Transfer Resume Register for CRU Image Data	AMnFIFOTRST	H’0164	R/W	H’0000_0000
AXI Master FIFO Pointer Register for CRU Image Data	AMnFIFOPNTR	H’0168	R	H’0100_0000
Reserved		H’016C-H’016F	—	—
Reserved		H’0170	—	—
AXI Master Transfer Stop Register for CRU Image Data	AMnAXISTP	H’0174	R/W	H’0000_0000
AXI Master Transfer Stop Status Register for CRU Image Data	AMnAXISTPACK	H’0178	R	H’0000_0000
Reserved		H’017C-H’017F	—	—
Reserved		H’0184-H’018F	—	—
Memory Bank 1 Base Address Lower Register for CRU Statistics Data	AMnSDMB1ADDRL	H’0190	R/W	H’0000_0000
Memory Bank 1 Base Address Higher Register for CRU Statistics Data	AMnSDMB1ADDRH	H’0194	R/W	H’0000_0000
Memory Bank 2 Base Address Lower Register for CRU Statistics Data	AMnSDMB2ADDRL	H’0198	R/W	H’0000_0000
Memory Bank 2 Base Address Higher Register for CRU Statistics Data	AMnSDMB2ADDRH	H’019C	R/W	H’0000_0000
Memory Bank 3 Base Address Lower Register for CRU Statistics Data	AMnSDMB3ADDRL	H’01A0	R/W	H’0000_0000
Memory Bank 3 Base Address Higher Register for CRU Statistics Data	AMnSDMB3ADDRH	H’01A4	R/W	H’0000_0000
Memory Bank 4 Base Address Lower Register for CRU Statistics Data	AMnSDMB4ADDRL	H’01A8	R/W	H’0000_0000
Memory Bank 4 Base Address Higher Register for CRU Statistics Data	AMnSDMB4ADDRH	H’01AC	R/W	H’0000_0000
Memory Bank 5 Base Address Lower Register for CRU Statistics Data	AMnSDMB5ADDRL	H’01B0	R/W	H’0000_0000
Memory Bank 5 Base Address Higher Register for CRU Statistics Data	AMnSDMB5ADDRH	H’01B4	R/W	H’0000_0000
Memory Bank 6 Base Address Lower Register for CRU Statistics Data	AMnSDMB6ADDRL	H’01B8	R/W	H’0000_0000
Memory Bank 6 Base Address Higher Register for CRU Statistics Data	AMnSDMB6ADDRH	H’01BC	R/W	H’0000_0000

Table 35.11 Register Configuration of the Image Processing Module (3/4)

Register Name	Register Symbol Name	Address	R/W	Initial Value
Memory Bank 7 Base Address Lower Register for CRU Statistics Data	AMnSDMB7ADDRL	H’01C0	R/W	H’0000_0000
Memory Bank 7 Base Address Higher Register for CRU Statistics Data	AMnSDMB7ADDRH	H’01C4	R/W	H’0000_0000
Memory Bank 8 Base Address Lower Register for CRU Statistics Data	AMnSDMB8ADDRL	H’01C8	R/W	H’0000_0000
Memory Bank 8 Base Address Higher Register for CRU Statistics Data	AMnSDMB8ADDRH	H’01CC	R/W	H’0000_0000
Memory Bank Enable Register for CRU Statistics Data	AMnSDMBVALID	H’01D0	R/W	H’0000_0001
Memory Bank Status Register for CRU Statistics Data	AMnSDMBS	H’01D4	R	H’0000_0000
AXI-SD Master Transfer Setting Register	AMnSDAXIATTR	H’01D8	R/W	H’0000_0150
Reserved		H’01DC-H’01DF	—	—
AXI-SD Master FIFO Setting Register	AMnSDFIFO	H’01E0	R/W	H’0000_0001
AXI-SD Master Transfer Re-Start	AMnSDFIFOTRST	H’01E4	R/W	H’0000_0000
AXI-SD Master FIFO Pointer Register	AMnSDFIFOPNTR	H’01E8	R	H’0000_0000
Reserved		H’01EC-H’01EF	—	—
Reserved		H’01F0	—	—
AXI-SD Master Transfer Stop Register	AMnSDAXISTP	H’01F4	R/W	H’0000_0000
AXI-SD Master Transfer Stop Status Register	AMnSDAXISTPACK	H’01F8	R	H’0000_0000
Reserved		H’01FC-H’01FF	—	—
Image processing				
CRU Image Processing Enable Register	ICnEN	H’0200	R/W	H’0000_0000
CRU Image Processing Register Setting Change Control Register	ICnREGC	H’0204	R/W	H’0000_0000
CRU Image Processing Main Control Register	ICnMC	H’0208	R/W	H’001E_30FE
CRU Image Clipping Start Line Register	ICnSLPrC	H’0210	R/W	H’0000_0000
CRU Image Clipping End Line Register	ICnELPrC	H’0214	R/W	H’0000_0000
CRU Image Clipping Start Pixel Register	ICnSPPrC	H’0218	R/W	H’0000_0000
CRU Image Clipping End Pixel Register	ICnEPPrC	H’021C	R/W	H’0000_0000
CRU Scan Line Interrupt Register	ICnSI	H’0220	R/W	H’0000_0000
CRU Linear Matrix Offset Register	ICnLMXOF	H’0224	R/W	H’0000_0000
CRU Linear Matrix R Coefficient 1 Register	ICnLMXRC1	H’0228	R/W	H’0000_0400
CRU Linear Matrix R Coefficient 2 Register	ICnLMXRC2	H’022C	R/W	H’0000_0000
CRU Linear Matrix G Coefficient 1 Register	ICnLMXGC1	H’0230	R/W	H’0000_0000
CRU Linear Matrix G Coefficient 2 Register	ICnLMXGC2	H’0234	R/W	H’0000_0400
CRU Linear Matrix B Coefficient 1 Register	ICnLMXBC1	H’0238	R/W	H’0000_0000
CRU Linear Matrix B Coefficient 2 Register	ICnLMXBC2	H’023C	R/W	H’0400_0000
CRU Statistics Control 1 Register	ICnSTIC1	H’0240	R/W	H’0000_0000
CRU Statistics Control 2 Register	ICnSTIC2	H’0244	R/W	H’0000_0000
Reserved		H’0248-H’024F	—	—
CRU Parallel I/F Control Register	ICnPIFC	H’0250	R/W	H’0000_0000
CRU Module Status Register	ICnMS	H’0254	R	H’0000_0000
CRU Frame Subsampling Control Register	ICnDEC	H’0258	R/W	H’0000_0000
CRU Line Count Register	ICnLC	H’025C	R	H’0000_0000
CRU Word Count Register	ICnWC	H’0260	R	H’0000_0000
CRU Word Count Check Control Register	ICnEWC	H’0264	R/W	H’0000_0000

Table 35.11 Register Configuration of the Image Processing Module (4/4)

Register Name	Register Symbol Name	Address	R/W	Initial Value
CRU Frame Subsampling Interrupt Control Register	ICnINTCTRL	H'0268	R/W	H'0000_0000
CRU Output Image Format Register	ICnDMR	H'026C	R/W	H'0000_0000
CRU YCbCr → RGB Color Space Conversion Coefficient 1 Register	ICnCSCC1	H'0270	R/W	H'0000_129F
CRU YCbCr → RGB Color Space Conversion Coefficient 2 Register	ICnCSCC2	H'0274	R/W	H'0100_0800
CRU YCbCr → RGB Color Space Conversion Coefficient 3 Register	ICnCSCC3	H'0278	R/W	H'1989_0D02
CRU YCbCr → RGB Color Space Conversion Coefficient 4 Register	ICnCSCC4	H'027C	R/W	H'0645_2045
CRU RGB → YCbCr Color Space Conversion Y Coefficient 1 Register	ICnYCCR1	H'0280	R/W	H'0000_0107
CRU RGB → YCbCr Color Space Conversion Y Coefficient 2 Register	ICnYCCR2	H'0284	R/W	H'0064_0204
CRU RGB → YCbCr Color Space Conversion Y Coefficient 3 Register	ICnYCCR3	H'0288	R/W	H'0A00_0100
CRU RGB → YCbCr Color Space Conversion Cb Coefficient 1 Register	ICnCBCCR1	H'028C	R/W	H'0000_1F68
CRU RGB → YCbCr Color Space Conversion Cb Coefficient 2 Register	ICnCBCCR2	H'0290	R/W	H'01C2_1ED6
CRU RGB → YCbCr Color Space Conversion Cb Coefficient 3 Register	ICnCBCCR3	H'0294	R/W	H'0A00_0800
CRU RGB → YCbCr Color Space Conversion Cr Coefficient 1 Register	ICnCRCCR1	H'0298	R/W	H'0000_01C2
CRU RGB → YCbCr Color Space Conversion Cr Coefficient 2 Register	ICnCRCCR2	H'029C	R/W	H'1FB7_1E87
CRU RGB → YCbCr Color Space Conversion Cr Coefficient 3 Register	ICnCRCCR3	H'02A0	R/W	H'0A00_0800
Reserved		H'02A4-H'02AF	—	—
CRU Lookup Table Control Register	ICnLUT	H'02B0	R/W	H'0000_0000
CRU Lookup Table Status Register	ICnLUTS	H'02B4	R	H'0000_0000
CRU Lookup Table Pointer Register	ICnLUTP	H'02B8	R/W	H'0000_0000
CRU Lookup Table Data Register	ICnLUTD	H'02BC	R/W	H'0000_0000
CRU Test Image Generation Control 1 Register	ICnTICTRL1	H'02C0	R/W	H'0000_0000
CRU Test Image Generation Control 2 Register	ICnTICTRL2	H'02C4	R/W	H'0000_0000
CRU Test Image Size Setting 1 Register	ICnTISIZE1	H'02C8	R/W	H'0000_0000
CRU Test Image Size Setting 2 Register	ICnTISIZE2	H'02CC	R/W	H'0000_0000
Reserved		H'02D0-H'03FF	—	—

Note: Addresses other than above are reserved. Do not access them.
Reserved areas should not be accessed.

35.2.3 Register Descriptions

Access the following registers only in units of 32 bits.

35.2.3.1 CRU Control Register (CRUnCTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VINSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	VINSEL	0b	RW	Video data input switch setting 0b: MIPI I/F 1b: Parallel I/F

35.2.3.2 CRU Interrupt Enable Register (CRUnIE)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	CEE	WIE	SIE	EFE	SFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SDFEOVWE	SDDECEE	SDSLVEE	SDFOE	—	—	—	—	FEOVWE	DECEE	SLVEE	FOE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28 to 21	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	CEE	0b	RW	ITU-R BT.656 DECERR interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
19	WIE	0b	RW	Work count value error interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
18	SIE	0b	RW	Scan line interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
17	EFE	0b	RW	Frame end interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
16	SFE	0b	RW	Frame start interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
15 to 12	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	SDFEOVWE	0b	RW	Frame End Overwrite (Statistics data-related) interrupt enable control 0b: No interrupts are generated 1b: Interrupts are generated
10	SDDECEE	0b	RW	AXI DECERR Error (Statistics data-related) interrupt enable control 0b: No interrupts are generated 1b: Interrupts are generated
9	SDSLVEE	0b	RW	AXI SLVERR Error (Statistics data-related) interrupt enable control 0b: No interrupts are generated 1b: Interrupts are generated
8	SDFOE	0b	RW	FIFO overflow (Statistics data-related) interrupt enable control 0b: No interrupts are generated 1b: Interrupts are generated
7 to 4	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
3	FEOVWE	0b	RW	Frame End Overwrite (image-related) interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
2	DECEE	0b	RW	DECERR (image-related) interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
1	SLVEE	0b	RW	SLVERR (image-related) interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.
0	FOE	0b	RW	FIFO overflow (image-related) interrupt control 0b: No interrupts are generated. 1b: Interrupts are generated.

35.2.3.3 CRU Interrupt Status Register (CRUnINTS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	CES	WIS	SIS	EFS	SFS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SDFEOVWS	SDDECES	SDSLVES	SDFOS	—	—	—	—	FEOVWS	DECES	SLVES	FOS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
28 to 21	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	CES	0b	RW	ITU-R BT.656 DECERR interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
19	WIS	0b	RW	Word count value error interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
18	SIS	0b	RW	Scan line interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
17	EFS	0b	RW	Frame end interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
16	SFS	0b	RW	Frame start interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	SDFEOVWS	0b	RW	Frame End Overwrite (Statistics data-related) interrupt status An error occurs if the end of the next frame is received while processing a frame end. Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
10	SDDECES	0b	RW	AXI DECERR Error (Statistics data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
9	SDSLVES	0b	RW	AXI SLVERR Error (Statistics data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
8	SDFOS	0b	RW	FIFO overflow (Statistics data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
7 to 4	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
3	FEOVWS	0b	RW	Frame End Overwrite (image-related) interrupt status An error occurs if the end of the next frame is received while processing a frame end. Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
2	DECES	0b	RW	DECERR (image-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
1	SLVES	0b	RW	SLVERR (image-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated
0	FOS	0b	RW	FIFO overflow (image-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated

35.2.3.4 CRU Reset Register (CRUnRST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VRESETN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	VRESETN	0b	RW	0b: Resets the image processing module. 1b: Releases the image processing module from the reset state.

35.2.3.5 CRU General Read/Write Register (CRUnCOM)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	COMMON[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COMMON[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	COMMON [31:0]	All 0	RW	General register

35.2.3.6 Memory Bank 1 Base Address (Lower) Register for CRU Image Data (AMnMB1ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB1ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB1ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB1ADDRL [31:9]	All 0	RW	Image data transfer destination base address 1 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H’000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.7 Memory Bank 1 Base Address (Higher) Register for CRU Image Data (AMnMB1ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB1ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB1ADDRH [1:0]	00b	RW	Image data transfer destination base address 1 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.8 Memory Bank 2 Base Address (Lower) Register for CRU Image Data (AMnMB2ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB2ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB2ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB2ADDRL [31:9]	All 0	RW	Image data transfer destination base address 2 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.9 Memory Bank 2 Base Address (Higher) Register for CRU Image Data (AMnMB2ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB2ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB2ADDRH [1:0]	00b	RW	Image data transfer destination base address 2 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.10 Memory Bank 3 Base Address (Lower) Register for CRU Image Data (AMnMB3ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB3ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB3ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB3ADDRL [31:9]	All 0	RW	Image data transfer destination base address 3 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H’000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.11 Memory Bank 3 Base Address (Higher) Register for CRU Image Data (AMnMB3ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB3ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB3ADDRH [1:0]	00b	RW	Image data transfer destination base address 3 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.12 Memory Bank 4 Base Address (Lower) Register for CRU Image Data (AMnMB4ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB4ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB4ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB4ADDRL [31:9]	All 0	RW	Image data transfer destination base address 4 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.13 Memory Bank 4 Base Address (Higher) Register for CRU Image Data (AMnMB4ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB4ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB4ADDRH [1:0]	00b	RW	Image data transfer destination base address 4 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.14 Memory Bank 5 Base Address (Lower) Register for CRU Image Data (AMnMB5ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB5ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB5ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB5ADDRL [31:9]	All 0	RW	Image data transfer destination base address 5 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.15 Memory Bank 5 Base Address (Higher) Register for CRU Image Data (AMnMB5ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB5ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB5ADDRH [1:0]	00b	RW	Image data transfer destination base address 5 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.16 Memory Bank 6 Base Address (Lower) Register for CRU Image Data (AMnMB6ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB6ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB6ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB6ADDRL [31:9]	All 0	RW	Image data transfer destination base address 6 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.17 Memory Bank 6 Base Address (Higher) Register for CRU Image Data (AMnMB6ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB6ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB6ADDRH [1:0]	00b	RW	Image data transfer destination base address 6 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.18 Memory Bank 7 Base Address (Lower) Register for CRU Image Data (AMnMB7ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB7ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB7ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB7ADDRL [31:9]	All 0	RW	Image data transfer destination base address 7 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.19 Memory Bank 7 Base Address (Higher) Register for CRU Image Data (AMnMB7ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB7ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB7ADDRH [1:0]	00b	RW	Image data transfer destination base address 7 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.20 Memory Bank 8 Base Address (Lower) Register for CRU Image Data (AMnMB8ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB8ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB8ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB8ADDRL [31:9]	All 0	RW	Image data transfer destination base address 8 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H’000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.21 Memory Bank 8 Base Address (Higher) Register for CRU Image Data (AMnMB8ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB8ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MB8ADDRH [1:0]	00b	RW	Image data transfer destination base address 8 (valid bits: [33:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.22 UV Data Address Offset (Lower) Register for CRU Image Data (AMnUVAOFL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UVAOFL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UVAOFL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	UVAOFL [31:9]	All 0	RW	UV data address offset when outputting image data YUV (valid bits: [31:9]) Set it in the base address in units of 512 bytes. *Note*: The lower nine bits cannot be written to (cannot be modified while the AXI is operating).
8 to 0	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.23 UV Data Address Offset (Higher) Register for CRU Image Data (AMnUVAOFH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UVAOFH[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	UVAOFH[2:0]	000b	RW	UV data address offset when outputting image data YUV (valid bits: [34:32]) (These bits cannot be modified while the AXI is operating.)

35.2.3.24 Memory Bank Enable Register for CRU Image Data (AMnMBVALID)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	MBVALID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	MBVALID[7:0]	H'01	RW	Image data transfer destination valid address (Details is written in **Section 35.2.4.3**.) [0]: 0: MB1 invalid, 1: MB1 valid [1]: 0: MB2 invalid, 1: MB2 valid [2]: 0: MB3 invalid, 1: MB3 valid [3]: 0: MB4 invalid, 1: MB4 valid [4]: 0: MB5 invalid, 1: MB5 valid [5]: 0: MB6 invalid, 1: MB6 valid [6]: 0: MB7 invalid, 1: MB7 valid [7]: 0: MB8 invalid, 1: MB8 valid *Note*: Set the valid destination address from the LSB continuously. Data transferred from the LSB is written sequentially only in valid banks, and the process is repeated from the start (LSB) when the last address is reached. (These bits cannot be modified while the AXI is operating.)

35.2.3.25 Memory Bank Status Register for CRU Image Data (AMnMBS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	MBSTS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	MBSTS[2:0]	000b	R	Indicates which memory bank the image data is being transferred to, MB1 to MB8. (H'0: MB1, H'1: MB2 … H'6: MB7, H'7: MB8)

35.2.3.26 AXI Master Transfer Setting Register for CRU Image Data (AMnAXIATTR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	AXILEN[3:0]			
Initial Value	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 24	—	H’0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 16	—	H’00	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13, 12	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11, 10	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	—	01b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	—	101b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	AXILEN[3:0]	H'0	RW	AXI burst max length setting for image data (these bits cannot be modified while the AXI is operating.) For YC separation mode (ICnDMR.YCMODE[2:0] = 010b): H'0: up to 1-beat burst H'1: up to 1-beat burst H'3: up to 2-beat burst H'7: up to 4-beat burst H'F: up to 8-beat burst Others: Setting prohibited. Except for YC separation mode (ICnDMR.YCMODE[2:0] ≠ 010b): H'0: up to 1-beat burst H'1: up to 2-beat burst H'3: up to 4-beat burst H'7: up to 8-beat burst H'F: up to 16-beat burst Others: Setting prohibited. To improve the transfer performance of this unit, it is recommended to use it in H'F.

35.2.3.27 AXI Master FIFO Setting Register for CRU Image Data (AMnFIFO)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FIFOOVFREC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	FIFOOVFREC	1b	RW	Recovery mode when image data FIFO overflowed The image-related AXI master stops when the image data FIFO overflowed. The operation after an overflow can be selected: 0b: Auto recovery: The CRU recovers automatically in the next frame after the image-related AXI master stopped and the FIFO was cleared. After recovery, transfer resumes to the start address of the MB to which data was being transferred to. 1b: Stop: The image-related AXI master stops and the FIFO is cleared. The status can be verified by the reg_axi_mst_err_int interrupt. The CRU recovers in the next frame after AXI_TRANS_START bit in the AMnFIFOTRST register is set to 1. The image data being processed at the time of the FIFO overflow will be corrupted. The FIFO overflow may have been caused by bus congestion. Take actions for bus congestion.

35.2.3.28 AXI Master Transfer Resume Register for CRU Image Data (AMnFIFOTRST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AXI_TRANS_START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	AXI_TRANS_START	0b	RW	AXI transfer control for image data (valid when AMnFIFO.FIFOOVFREC = 1). See **Section 35.2.3.27**. 0b: Writing prohibited. Write 1 to this bit to recover the image-related AXI (transfer resume) (This bit returns to 0 when transfer is resumed.)

35.2.3.29 AXI Master FIFO Pointer Register for CRU Image Data (AMnFIFOPNTR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFORPNTR_UV[7:0]								FIFORPNTR_Y[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	FIFOWPNTR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	FIFORPNTR_UV[7:0]	H'01	R	FIFO read pointer for image data (for UV) Not during YC separation mode (ICnDMR.YCMODE[2:0] = 010b): Unused During YC separation mode (ICnDMR.YCMODE[2:0] = 010b): FIFO read pointer for UV data [7:1] RAM read pointer value for UV data FIFO [0] Fixed to 1
23 to 16	FIFORPNTR_Y[7:0]	H'00	R	FIFO read pointer for image data Not during YC separation mode (ICnDMR.YCMODE[2:0] = 010b): All FIFO read pointer [7:0] RAM read pointer value for FIFO During YC separation mode (ICnDMR.YCMODE[2:0] = 010b): Y data FIFO read pointer [7:1] RAM read pointer value for Y data FIFO [0] Fixed to 0
15 to 8	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	FIFOWPNTR[7:0]	H'00	R	FIFO write pointer for image data Not during YC separation mode (ICnDMR.YCMODE[2:0] = 010b): [7:0] RAM write pointer value for FIFO *Note*: The amount of data stored in the FIFO can be found by FIFOWPNTR[7:0] - FIFORPNTR_Y[7:0]. During YC separation mode (ICnDMR.YCMODE[2:0] = 010b): [7:1] RAM write pointer value for FIFO [0] Fixed to 0 *Note*: The amount of data stored in the FIFO can be found by FIFOWPNTR[7:1] - FIFORPNTR_Y[7:1], FIFOWPNTR[7:1] - FIFORPNTR_UV[7:1].

35.2.3.30 AXI Master Transfer Stop Register for CRU Image Data (AMnAXISTP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AXI_STOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	AXI_STOP	0b	RW	Image data AXI transfer stop (used to stop the CRU from receiving data. Perform the reception stop procedure described in **Section 35.3.2** and **Section 35.3.5**.) 0b: Does not stop the AXI master from transferring data. 1b: Stops the AXI master from transferring data.

35.2.3.31 AXI Master Transfer Stop Status Register for CRU Image Data (AMnAXISTPACK)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AXI_STOP_ACK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	AXI_STOP_ACK	0b	R	Image data AXI transfer stop response (used to stop the CRU from receiving data. Perform the reception stop procedure described in **Section 35.3.2** and **Section 35.3.5**.) 0b: The AXI master is transferring data. 1b: The AXI master has finished transferring data (valid when AXI_STOP = 1)

35.2.3.32 Memory Bank 1 Base Address Lower Register for CRU Statistics Data (AMnSDMB1ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB1ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB1ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB1ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 1 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.33 Memory Bank 1 Base Address Higher Register for CRU Statistics Data (AMnSDMB1ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB1ADDRH [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB1ADDRH[1:0]	00b	RW	Statics data transfer destination base address 1 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.34 Memory Bank 2 Base Address Lower Register for CRU Statistics Data (AMnSDMB2ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB2ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB2ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB2ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 2 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.35 Memory Bank 2 Base Address Higher Register for CRU Statistics Data (AMnSDMB2ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB2ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB2ADDRH[1:0]	00b	RW	Statics data transfer destination base address 2 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.36 Memory Bank 3 Base Address Lower Register for CRU Statistics Data (AMnSDMB3ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB3ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB3ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB3ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 3 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.37 Memory Bank 3 Base Address Higher Register for CRU Statistics Data (AMnSDMB3ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB3ADDRH [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB3ADDRH[1:0]	00b	RW	Statics data transfer destination base address 3 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.38 Memory Bank 4 Base Address Lower Register for CRU Statistics Data (AMnSDMB4ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB4ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB4ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB4ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 4 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.39 Memory Bank 4 Base Address Higher Register for CRU Statistics Data (AMnSDMB4ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB4ADDRH [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB4ADDRH[1:0]	00b	RW	Statics data transfer destination base address 4 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.40 Memory Bank 5 Base Address Lower Register for CRU Statistics Data (AMnSDMB5ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB5ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB5ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB5ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 5 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.41 Memory Bank 5 Base Address Higher Register for CRU Statistics Data (AMnSDMB5ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB5ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB5ADDRH[1:0]	00b	RW	Statics data transfer destination base address 5 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.42 Memory Bank 6 Base Address Lower Register for CRU Statistics Data (AMnSDMB6ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB6ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB6ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB6ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 6 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.43 Memory Bank 6 Base Address Higher Register for CRU Statistics Data (AMnSDMB6ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB6ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB6ADDRH[1:0]	00b	RW	Statics data transfer destination base address 6 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.44 Memory Bank 7 Base Address Lower Register for CRU Statistics Data (AMnSDMB7ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB7ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB7ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB7ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 7 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.45 Memory Bank 7 Base Address Higher Register for CRU Statistics Data (AMnSDMB7ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB7ADDRH [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB7ADDRH[1:0]	00b	RW	Statics data transfer destination base address 7 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.46 Memory Bank 8 Base Address Lower Register for CRU Statistics Data (AMnSDMB8ADDRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB8ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB8ADDRL[15:9]							—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB8ADDRL[31:9]	All 0	RW	Statistics data transfer destination base address 8 (valid bits are [31:9]) Set it as the base address in units of 512 bytes. *Note*: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-SD is operating.)
8 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

35.2.3.47 Memory Bank 8 Base Address Higher Register for CRU Statistics Data (AMnSDMB8ADDRH)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDMB8ADDRH[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	SDMB8ADDRH[1:0]	00b	RW	Statics data transfer destination base address 8 (valid bits are [33:32]) *Note*: These bits cannot be modified while the AXI-SD is operating.

35.2.3.48 Memory Bank Enable Register for CRU Statistics Data (AMnSDMBVALID)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SDMBVALID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	SDMBVALID[7:0]	H'01	RW	Statistics data transfer destination valid address (the details are described in **Section 35.2.4.3, Transfer to DRAM.** [0]: 0b: SDMB1 invalid, 1b: SDBM1 valid [1]: 0b: SDMB2 invalid, 1b: SDBM2 valid [2]: 0b: SDMB3 invalid, 1b: SDBM3 valid [3]: 0b: SDMB4 invalid, 1b: SDBM4 valid [4]: 0b: SDMB5 invalid, 1b: SDBM5 valid [5]: 0b: SDMB6 invalid, 1b: SDBM6 valid [6]: 0b: SDMB7 invalid, 1b: SDBM7 valid [7]: 0b: SDMB8 invalid, 1b: SDBM8 valid *Note*: Set the valid destination address from the LSB continuously. Data transferred from the LSB are written sequentially only to valid memory banks, and the processing is repeated from the start (LSB) when the last address is reached. (These bits cannot be modified while AXI-SD is operating.)

35.2.3.49 Memory Bank Status Register for CRU Statistics Data (AMnSDMBS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	SDBMSTS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	SDMBSTS [2:0]	000b	R	Indicates which memory bank from SDMB1 to SDMB8 the statistics data are being transferred to. 000b: SDMB1 001b: SDMB2 010b: SDMB3 011b: SDMB4 100b: SDMB5 101b: SDMB6 110b: SDMB7 111b: SDMB8

35.2.3.50 AXI-SD Master Transfer Setting Register (AMnSDAXIATTR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SDAXILEN[3:0]			
Initial Value	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 24	—	H'0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 16	—	H'00	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13, 12	—	00b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11, 10	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9, 8	—	01b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	—	101b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3 to 0	SDAXILEN [3:0]	H'0	RW	Maximum value setting of AXI-SD burst length for statistics data (changing the value during AXI-SD operation is prohibited) 0000b: Up to 1 burst 0001b: Up to 2 burst 0010b: Up to 3 burst 0011b: Up to 4 burst 0100b: Up to 5 burst 0101b: Up to 6 burst ⋮ ⋮ 1111b: Up to 16 burst

35.2.3.51 AXI-SD Master FIFO Setting Register (AMnSDFIFO)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDFIFOOVFREC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SDFIFOOVFREC	1b	RW	Recovering mode setting for Statics Data FIFO-SD overflow When the line number being processed in a frame of the image processing module matches the line setting value specified by this bit, an image_conv_int interrupt can be generated. 0b: Auto recovering After AXI Mater stop and FIFO-SD clear, recover at next frame automatically. 1b: Manual recovering (Debugging mode) After AXI Master stop and FIFO-SD clear, recover at next frame by setting 1 to AMnSDFIFOTRST.SDAXI_TRANS_START.

35.2.3.52 AXI-SD Master Transfer Re-Start (AMnSDFIFOTRST)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDAXI_TRANS_START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SDAXI_TRANS_START	0b	RW	Statistics Data AXI transfer control, this bit is effective when AMnSDFIFO.SDFIFOOVFREC is 1. Details, refer to **Section 35.2.3.51, AXI-SD Master FIFO Setting Register (AMnSDFIFO)**. 0b: Prohibited to write 0 (0 is set automatically, after recovering transfer). 1b: Writing 1 for AXI-SD recover (recovering transfer).

35.2.3.53 AXI-SD Master FIFO Pointer Register (AMnSDFIFOPNTR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	SDFIFORPNTR[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SDFIFOWPNTR[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20 to 16	SDFIFORPNTR[4:0]	H'00	R	Statistics Data FIFO-SD read pointer for statistics data The read pointer and write pointer comparison allows checking if there are data in the FIFO-SD. FIFO-SD read pointer [3:0] Real RAM address for FIFO-SD (Max: H'F) *Note 1.* When (AMnSDFIFOPNTR.SDFIFOWPNTR[3:0] ≠ AMnSDFIFOPNTR.SDFIFORPNTR[3:0], data are present in FIFO-SD. *Note 2.* Refer to bit-3 to bit-0, bit-4 is only for debugging.
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	SDFIFOWPNTR[4:0]	H'00	R	Statistics Data FIFO-SD write pointer for statistics data The read pointer and write pointer comparison allows checking if there are data in the FIFO-SD. FIFO-SD write pointer [3:0] Real RAM address for FIFO-SD (Max: H'F) *Note 1.* When (AMnSDFIFOPNTR.SDFIFOWPNTR[3:0] ≠ AMnSDFIFOPNTR.SDFIFORPNTR[3:0], data are present in FIFO-SD. *Note 2.* Refer to bit-3 to bit-0, bit-4 is only for debugging.

35.2.3.54 AXI-SD Master Transfer Stop Register (AMnSDAXISTP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDAXISTOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SDAXISTOP	0b	RW	Statistics Data AXI-SD transfer stop (unused to stop the CRU receiving data. Follow the procedure for stopping reception described in **Section 35.3.2, Stopping Reception for the MIPI CSI-2 Input**. 0b: Disable to stop the AXI-SD mater transferring data. 1b: Enable to stop the AXI-SD master transferring data.

35.2.3.55 AXI-SD Master Transfer Stop Status Register (AMnSDAXISTPACK)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SDAXISTOPACK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	SDAXISTOPACK	0b	R	Statistics data AXI-SD transfer stop response (unused to stop CRU receiving data. Follow the procedure or stopping reception described in **Section 35.3.2, Stopping Reception for the MIPI CSI-2 Input**. 0b: The AXI-SD master is transferring data. 1b: The AXI-SD master has finished transferring data (valid when AXI_STOP = 1).

35.2.3.56 CRU Image Processing Enable Register (ICnEN)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ICEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	ICEN	0b	RW	Enables the operation of the image processing module. (Enable or disable the operation according to **Section 35.3.1** and **Section 35.3.2**.) 0b: Disables the image processing module. 1b: Enables the image processing module. *Note*: If this bit is set to 0, a frame start is not detected. If this bit is changed from 1 to 0 while a frame is being processed, the data at that time is processed as the frame end and then the next frames are not processed.

35.2.3.57 CRU Image Processing Register Setting Change Control Register (ICnREGC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	REFEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	REFEN	0b	RW	Allows register setting changes to be reflected. By using this register for an MIPI CSI-2 input, the registers at addresses H'1083_0208 to H'1083_0244 can be changed through frame synchronization during operation. (Allow or disallow this according to **Section 35.3.3.1**, **Section 35.3.3.2**.) 0b: Does not allow register setting changes to be reflected. 1b: Allows register setting changes to be reflected. *Note 1.* If this bit is set to 1 while the image processing module is operating (ICnEN.ICEN = 1), the register setting values (at addresses H'1083_0208 to H'1083_0244) are reflected at a frame start, and then the bit is cleared to 0 automatically. *Note 2.* If the image processing module is stopped (ICnEN.ICEN = 0), the register setting values can be reflected irrespective of this register. Note that registers other than the above cannot be changed during operation.

35.2.3.58 CRU Image Processing Main Control Register (ICnMC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	RAWSTTYP[1:0]		VCSEL[1:0]		INF[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	IBINSEL	CLP[1:0]		—	—	—	—	STITHR	LUTTHR	CSCTHR	LMXTHR	DEMTHR	CLPTHR	DECTHR	—
Initial Value	0	0	1	1	0	0	0	0	1	1	1	1	1	1	1	0
R/W	R	RW	RW	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored..
25, 24	RAWSTTYP [1:0]	00b	RW	Specify the type of the 1st pixel of the 1st line of the RAW data This setting is for recognizing the components of RAW data when demosaicing. 00b: R (RGRG…) 01b: Gr (GRGR…) 10b: Gb (GBGB…) 11b: B (BGBG…)
23, 22	VCSEL[1:0]	00b	RW	Specify a virtual channel to be processed by the image processing module. A single channel specified by the register is selected from virtual channels 0 to 3.
21 to 16	INF[5:0]	H'1E	RW	Specify the MIPI input image formats to be processed in the image processing module. For details about the formats, see Data Type Codes in the MIPI CSI-2 V2.1 Specification. From the formats specified here, only one format is processed. *Note 1.* Initial value: YUV422 8-bit *Note 2.* The following values can be set. Other settings are prohibited. H'13 to H'16: Generic Long Packet Data Type H'18 to H'1A, H'1C to H'1F: YUV Data H'20 to H'24: RGB Data H'28 to H'2F: RAW Data H'30 to H'37: User-Defined Byte-based Data The data types specified here must have been set in CSI2nDTEL.DTEN and CSI2DTEH.DTEN of MIPI CSI2.
15	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
14	IBINSEL	0b	RW	Binary selection for UV of input image format YUV422 0b: Offset binary (straight binary) 1b: Two's complement binary

Bit	Bit Name	Initial Value	R/W	Description
13, 12	CLP[1:0]	11b	RW	Data clipping setting for the YCbCr (YUV) format To make the input data to be RGB-converted comply with the ITU-R BT.601 standard, specify whether to clip the values below 16 and above 240 (when YCbCr is 8-bit). 00b: No clipping for Y When CbCr is 8-bit: Values below 16 and above 240 are clipped for CbCr When CbCr is 10-bit: Values below 64 and above 960 are clipped for CbCr 01b: When YCbCr is 8-bit: Values below 16 and above 240 are clipped for Y Values below 16 and above 240 are clipped for CbCr When YCbCr is 10-bit: Values below 64 and above 960 are clipped for Y Values below 64 and above 960 are clipped for CbCr 10b: No clipping for Y When CbCr is 8-bit: Values below 16 are increased to 128 and values above 240 are clipped to 128 for CbCr When CbCr is 10-bit: Values below 64 are increased to 512 and values above 960 are clipped to 512 for CbCr 11b: No clipping for Y and CbCr (initial value)
11 to 8	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7	STITHR	1b	RW	Bypassing Statistics processing for RAW data 0: Statistics processing is performed 1: Statistics processing is not performed (bypassed) *Note*: This setting of 0b applies only for the corresponding format shown in **Table 35.12**.
6	LUTTHR	1b	RW	Bypassing LUT-based image color conversion Regardless of whether this setting is 0b or 1b, the output of the LUT is 8 bits for 10 bits of input. 0b: Image color conversion is performed. 1b: Image color conversion is not performed (bypassed). *Note*: This setting of 0b applies only for the corresponding format shown in **Table 35.12**.
5	CSCTHR	1b	RW	Bypassing color space conversion (YCbCr(YUV) → RGB, RGB → YCbCr(YUV)) and data clipping. 0b: Color space conversion and data clipping are performed. 1b: Color space conversion and data clipping are not performed (bypassed). *Note 1.* This setting of 0b applies only for the corresponding format shown in **Table 35.12**. For other image formats, color space conversion is not performed regardless of the settings (RAW is subject to RGB conversion after demosaic processing). *Note 2.* Even when this bit is 1, the data is expanded from 8-bit to 10-bit.
4	LMXTHR	1b	RW	Bypassing Linear Matrix processing for RAW data 0: Linear Matrix processing is performed 1: Linear Matrix processing is not performed (bypassed) *Note*: This setting of 0b applies only for the corresponding format shown in **Table 35.12**. For other image formats, Linear Matrix processing is not performed regardless of the settings (RAW is subject to RGB conversion after demosaic processing).

Bit	Bit Name	Initial Value	R/W	Description
3	DEMTHR	1b	RW	Bypassing Demosaicing processing for RAW data 0: Demosaicing processing is performed 1: Demosaicing processing is not performed (bypassed) *Note*: This setting of 0b applies only for the corresponding format shown in **Table 35.12**. For other image formats, Demosaicing processing is not performed regardless of the settings (RAW is subject to RGB conversion after demosaic processing).
2	CLPTHR	1b	RW	Bypassing image clipping (Details is described in **Section 35.2.4.8**.) 0b: Image clipping is performed. 1b: Image clipping is not performed (bypassed). *Note*: This setting of 0b applies only for the corresponding format shown in **Table 35.12**.
1	DECTHR	1b	RW	Bypassing frame skip (Details is described in **Section 35.2.4.9**.) 0b: Frame subsampling is performed. 1b: Frame subsampling is not performed (bypassed). *Note*: This setting of 0b applies only for the corresponding format shown in **Table 35.12**.
0	—	0b	R	Reserved Whenever it is read, 0 is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

(1) Bypass Setting for Each Function

Figure 35.11 shows where the bypass settings by the DECTHR, CLPTHR, DEMTHR, LMXTHR, CSCTHR, LUTTHR, and STITHR bits of the ICnMC register work.

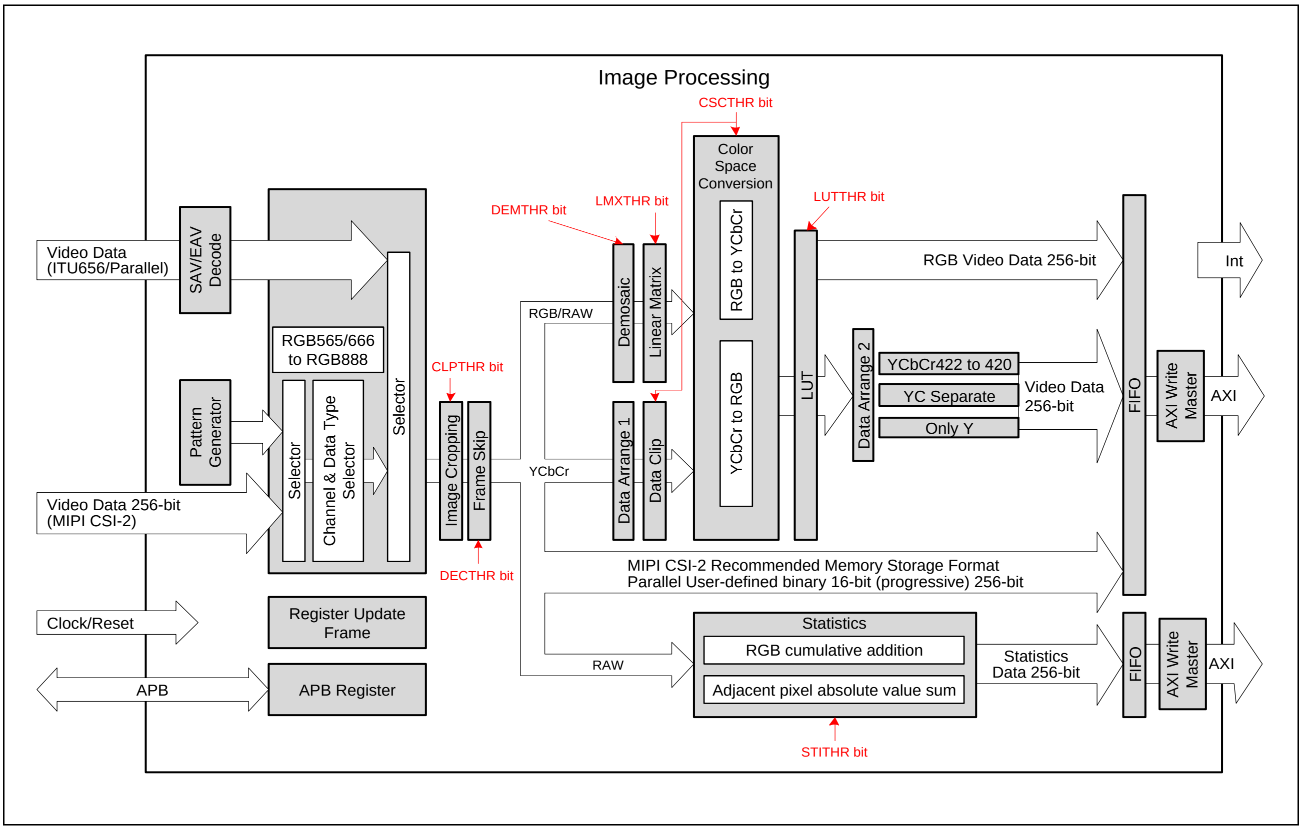


Figure 35.11 Bypass Setting of Each Function

35.2.3.59 CRU Image Clipping Start Line Register (ICnSLPrC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SLPrC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	SLPrC[11:0]	H’000	RW	Start line in the vertical direction when clipping the input image to the specified area Set a value of 1 or greater because the frame start is processed as the first line. The image is output from the specified line. (If 0 is specified, the same operation as for setting of 1 is performed.) *Note*: Set a value in a way that 1 ≤ ICnSLPrC.SLPrC ≤ ICnELPrC.ELPrC.

35.2.3.60 CRU Image Clipping End Line Register (ICnELPrC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ELPrC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored..
11 to 0	ELPrC[11:0]	H’000	RW	End line in the vertical direction when clipping the input image to the specified area Set a value of 1 or greater because the frame start is processed as the first line. The image is output up to the specified line. *Note*: Set a value in a way that 1 ≤ ICnSLPrC.SLPrC ≤ ICnELPrC.ELPrC.

35.2.3.61 CRU Image Clipping Start Pixel Register (ICnSPPrC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SPPrC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	SPPrC[11:0]	H’000	RW	Start pixel in the horizontal direction when clipping the input image to the specified area Set a value of 1 or greater because the line start is processed as the first pixel. The image is output from the specified pixel data. (If 0 is specified, the same operation as for setting of 1 is performed.) *Note 1.* Set a value in a way that 1 ≤ ICnSPPrC.SPPrC ≤ (ICnEPPrC.EPPrC + 1 − 320). *Note 2.* Specify an odd value when the input format is YUV422 (ICnMC.INF[5:0] = H’1E to H’1F or ICnPIFC.PINF[3:0] = H’0 to H’B). If an even number of 2 or greater is specified, the CRU operates as if an odd number (the specified number - 1) is specified. (Example: When 34 is specified, the data is output from the 33rd pixel.) *Note 3.* Even when the input format is RGB, etc., set a value that satisfies the following formula so that a line consists of an even number of pixels if the RGBtoYCbCr output is YUV422/420. ICnEPPrC.EPPrC - ICnSPPrC.SPPrC = 2N − 1 (N = 160, 161, 162,...)

35.2.3.62 CRU Image Clipping End Pixel Register (ICnEPPrC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	EPPrC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	EPPrC[11:0]	H’000	RW	End pixel in the horizontal direction when clipping the input image to the specified area Set a value of 1 or greater because the line start is processed as the first pixel. The image is output up to the specified image data. *Note 1.* Set a value in a way that 1 ≤ ICnSPPrC.SPPrC ≤ (ICnEPPrC.EPPrC + 1 − 320). *Note 2.* Specify an even value when the input forma is YUV422 (ICnMC.INF[5:0] = H’1E to H’1F or ICnPIFC.PINF[3:0] = H’0 to H’B). If an odd number is specified, the CRU operates as if an even number (the specified number + 1) is specified. (Example: When 127 is specified, the data is output up to the 128th pixel.) *Note 3.* Even when the input format is RGB, etc., set a value that satisfies the following formula so that a line consists of an even number of pixels if the RGBtoYCbCr output is YUV422/420. ICnEPPrC.EPPrC - ICnSPPrC.SPPrC = 2N − 1 (N = 160, 161, 162,...)

35.2.3.63 CRU Scan Line Interrupt Register (ICnSI)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SI[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	SI[11:0]	H'000	RW	Line specification for generating a scan line interrupt When the line number being processed in a frame of the image processing module matches the line setting value specified by this bit, an image_conv_int interrupt can be generated. *Note*: By using the ICnINTCTRL.DECINTE bit, whether to generate interrupts in subsampled frames can be specified.

35.2.3.64 CRU Linear Matrix Offset Register (ICnLMXOF)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	BOF[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GOF[7:0]								ROF[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	BOF[7:0]	H'00	RW	B offset value of linear matric calculation (bit 7: sign) *Note*: Setting range: −128 (H'80) to 127 (H'7F)
15 to 8	GOF[7:0]	H'00	RW	G offset value of linear matric calculation (bit7: sign) *Note*: Setting range: −128 (H'80) to 127 (H'7F)
7 to 0	ROF[7:0]	H'00	RW	R offset value of linear matric calculation (bit7: sign) *Note*: Setting range: −128 (H'80) to 127 (H'7F)

35.2.3.65 CRU Linear Matrix R Coefficient 1 Register (ICnLMXRC1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RR[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	RR[12:0]	H’0400	RW	Coefficient R of R data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H’1000) to 4095 (H’0FFF)

35.2.3.66 CRU Linear Matrix R Coefficient 2 Register (ICnLMXRC2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	RB[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RG[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	RB[12:0]	All 0	RW	Coefficient B of R data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H'1000) to 4095 (H'0FFF)
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	RG[12:0]	All 0	RW	Coefficient G of R data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H'1000) to 4095 (H'0FFF)

35.2.3.67 CRU Linear Matrix G Coefficient 1 Register (ICnLMXGC1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	GR[12:0]	All 0	RW	Coefficient R of G data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H’1000) to 4095 (H’0FFF)

35.2.3.68 CRU Linear Matrix G Coefficient 2 Register (ICnLMXGC2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	GB[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GG[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	GB[12:0]	All 0	RW	Coefficient B of G data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H'1000) to 4095 (H'0FFF)
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	GG[12:0]	H'0400	RW	Coefficient G of G data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H'1000) to 4095 (H'0FFF)

35.2.3.69 CRU Linear Matrix B Coefficient 1 Register (ICnLMXBC1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BR[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	RR[12:0]	All 0	RW	Coefficient R of B data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H’1000) to 4095 (H’0FFF)

35.2.3.70 CRU Linear Matrix B Coefficient 2 Register (ICnLMXBC2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	BB[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BG[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	BB[12:0]	H’0400	RW	Coefficient B of B data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H’1000) to 4095 (H’0FFF)
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	BG[12:0]	All 0	RW	Coefficient G of B data for linear matrix calculation (bit 12: sign) *Note*: Specify a signed 13-bit integer by multiplying the desired coefficient value (−4 to approx. 4) by 1024. Setting range: −4096 (H’1000) to 4095 (H’0FFF)

35.2.3.71 CRU Statistics Control 1 Register (ICnSTIC1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	STSADPOS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STUNIT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19 to 16	STSADPOS [3:0]	H'0	RW	Input data bit position selection (0 – 8) for the sum of absolute values for adjacent pixels for statistics processing. Use the 8 lower-order bits from the MSB-“n” bit for each type of RAW. *Note*: Comply with the restrictions described in ICnSTIC1.STUNIT.
15 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1, 0	STUNIT[1:0]	00b	RW	Setting an arithmetic processing unit for statistics processing (blocks). 00b: 16 × 16 pixels 01b: 32 × 32 pixels 10b: 64 × 64 pixels 11b: 128 × 128 pixels Example) In case of QVGA, STUNIT should be only 0. In case of VGA, STUNIT should be lesser than 1. In case of Full-HD, STUNIT should be lesser than 2. *Note*: Set the STUNIT and the STHPOS parameters, complying the following formula. ((HSIZE – STHPOS) >> (4 + STUNIT)) × (VSIZE >> (4 + STUNIT)) × 4 > 512 HIZE: horizontal pixels / VSIZE: vertical lines

35.2.3.72 CRU Statistics Control 2 Register (ICnSTIC2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	STHPOS[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8 to 0	STHPOS[8:0]	H’000	RW	Horizontal starting position for statistics processing (0 – 376 (H’178)). *Note 1.* Set the value of STUNIT and STHPOS so that there are 9 blocks or more per screen. *Note 2.* Normally 0 is used, and if the horizontal size of the pixel is 2800, it is assumed to be set to 376 (H’178).

35.2.3.73 CRU Parallel I/F Control Register (ICnPIFC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ENPOL	VSPOL	EC	—	ITL[2:0]			—	—	—	—	PINF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14	ENPOL	0b	RW	Polarity of the CAM_HREF input 0b: CAM_HREF signal is input with the polarity of High during the valid period of image data. 1b: CAM_HREF signal is input with the polarity of Low during the valid period of image data.
13	VSPOL	0b	RW	Polarity of the CAM_VSYNC input 0b: CAM_VSYNC signal is input with the polarity of VBLANK period High. 1b: CAM_VSYNC signal is input with the polarity of VBLANK period Low.
12	EC	0b	RW	Error correction according to ITU-R BT.656 0b: Error correction is not performed. 1b: Error correction is performed by using parity bits.
11	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10 to 8	ITL[2:0]	000b	RW	Progressive/interlace setting Bit 1: 0 x0b: Progressive input 01b: Interlaced input - no conversion 11b: Setting prohibited Bit 2 0b: Input the Top (Odd) field in field 1 in interlaced mode. 1b: Input the Bottom (Even) field in field 1 in interlaced mode. *Note 1.* The internal processing of the image processing module assumes that the Top field is input to field 1. Therefore, when 101b is set, field 1 and field 2 are swapped in the CRU. *Note 2.* The order of writing to the memory banks changes depending on the CAM_FIELD pin input and the ITL[2] setting. See **Table 35.15** for details. *Note 3.* For ITU-R BT.656, ITL[2:0] shall be set to 001b. Also, be sure to input from field1 (F = 0) at the beginning.
7 to 4	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	PINF[3:0]	H'0	RW	Format of the parallel I/F input 0000b: ITU-R BT.656 (8-bit) 0001b: ITU-R BT.656 (10-bit) 0010b: YCbCr separated 8-bit Cb1st except ITU-R BT.656 0011b: YCbCr separated 8-bit Cr1st except ITU-R BT.656 0100b: YCbCr multiplexed 8-bit Cb1st except ITU-R BT.656 0101b: YCbCr multiplexed 8-bit Cr1st except ITU-R BT.656 0110b: YCbCr multiplexed 8-bit Cb2nd except ITU-R BT.656 0111b: YCbCr multiplexed 8-bit Cr2nd except ITU-R BT.656 1000b: YCbCr multiplexed 10-bit Cb1st except ITU-R BT.656 1001b: YCbCr multiplexed 10-bit Cr1st except ITU-R BT.656 1010b: YCbCr multiplexed 10-bit Cb2nd except ITU-R BT.656 1011b: YCbCr multiplexed 10-bit Cr2nd except ITU-R BT.656 Others: Binary 16-bit except ITU-R BT.656

(1) Interlaced Mode

By setting the ITL[2] bit, the Top Field and the Bottom Field can be switched.

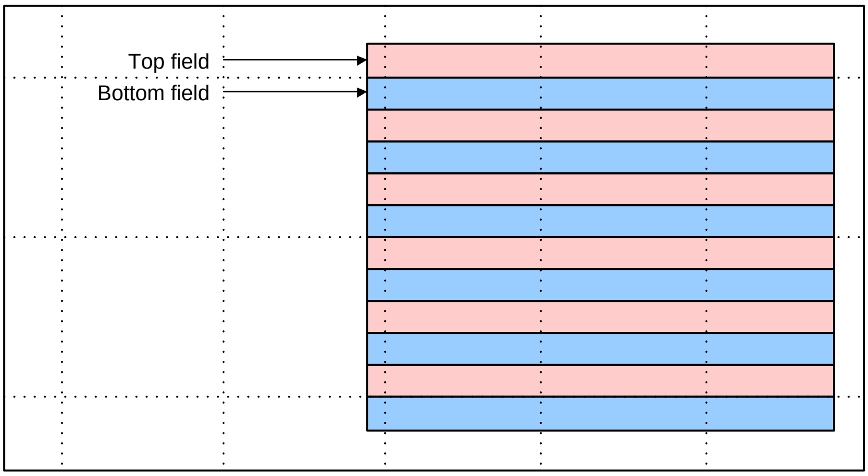


Figure 35.12 Interlaced Mode

35.2.3.74 CRU Module Status Register (ICnMS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	IA	AV	CA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	IA	0b	R	Indicates whether the image converter block (module covering the video data input up to the pre-FIFO stage) is processing. 0b: Image data has not been received yet. 1b: Image data is being converted.
1	AV	0b	R	Indicates whether processing is performed within the image-clipped area. (For details, see **Section 35.2.4.8**.) 0b: Processing is performed outside the image-clipped area. 1b: Processing is performed within the image-clipped area.
0	CA	0b	R	Indicates whether the frame is subsampled. (For details, see **Section 35.2.4.9**.) 0b: A subsampled frame is being processed. 1b: A frame which is not subsampled is being processed. *Note*: This bit is 0 when CRUnRST.VRESETN is 0. This bit is set to 1 two cycles of PCLK after CRUnRST.VRESETN is changed from 0 to 1.

35.2.3.75 CRU Frame Subsampling Control Register (ICnDEC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	FRMDEC[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	FRMDEC[3:0]	H'0	RW	Subsampling interval in frame subsampling (For details, see **Section 35.2.4.9**.) H'0: All images are passed. H'1: One pass every two frames H'2: One pass every three frames Omitted H'14: One pass every 15 frames H'15: One pass every 16 frames

35.2.3.76 CRU Line Count Register (ICnLC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	LC[11:0]	H’000	R	Indicate the line counter value before image clipping and subsampling in the image processing module.

35.2.3.77 CRU Word Count Register (ICnWC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	WCLC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	WC[15:0]	H’0000	R	Indicate the current word count from LINK for the MIPI CSI-2 input. *Note 1.* These bits are cleared to 0 at a frame start. *Note 2.* If the word count value does not match the value set in ICnEWC.EWC[15:0], the unmatched value is retained until the next frame start.
15 to 12	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	WCLC[11:0]	H’000	R	Indicate the count of continuous lines for which the word count value from LINK for the MIPI CSI-2 input matches the value set in ICnEWC.EWC[15:0]. *Note 1.* This value is cleared by a frame start. *Note 2.* If the word count value does not match the value set in ICnEWC.EWC[15:0], the unmatched value is retained until the next frame start.

35.2.3.78 CRU Word Count Check Control Register (ICnEWC)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EWC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	H'0000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 0	EWC[15:0]	H'0000	RW	Specify an expected word count value that is output from LINK for the MIPI CSI-2 input. If this setting value does not match the word count value that is output from LINK for the MIPI CSI-2 input, the image_conv_err_int interrupt can be output. These bits are used for debugging when the operation for the MIPI CSI-2 input is abnormal.

35.2.3.79 CRU Frame Subsampling Interrupt Control Register (ICnINTCTRL)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DECINTE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	DECINTE	0b	RW	Specifies whether an interrupt (CRUnINTS.SFS, CRUnINTS.EFS, or CRUnINTS.SI) is generated for a subsampled frame. 0b: No interrupt is generated for a subsampled frame. (Initial value) 1b: An interrupt is generated for a subsampled frame.

35.2.3.80 CRU Output Image Format Register (ICnDMR)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	A8BIT[7:0]								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OBINSEL	—	YCMODE[2:0]			—	—	RGBMODE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	A8BIT[7:0]	H'00	RW	Alpha bit Alpha value in the ARGB8888 format output
23 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	OBINSEL	0b	RW	Output binary selection for YCbCr data 0b: Offset binary (straight binary) 1b: Two's complement binary *Note*: Valid when the output is YUV422/420.
7	—	0b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
6 to 4	YCMODE[2:0]	000b	RW	YC data output format selection 000b: YCbCr422, 8-bit multiplexed (YUYV format (Y 1st)) 001b: YCbCr422, 8-bit multiplexed (UYVY format (Cb 1st)) 010b: YC separation from YCbCr422 011b: Y component extraction from YCbCr422 100b: YCbCr420, 8-bit multiplexed (YUYV format (Y 1st)) 101b: YCbCr420, 8-bit multiplexed (UYVY format (Cb 1st)) 110b: Setting prohibited 111b: Setting prohibited *Note*: YCbCr420 input is only supported for interlaced (parallel input), YCbCr420 output is not supported YC separation.
3 to 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	RGBMODE [1:0]	00b	RW	Format selection for RGB data output 00b: RGB888 (24-bit) 01b: RGB888 (32-bit) (The upper eight bits are padded with H'00.) 10b: ARBG888 (32-bit, B First A Last) 11b: ARBG888 (32-bit, A First B Last)

35.2.3.81 CRU YCbCr → RGB Color Space Conversion Coefficient 1 Register (ICnCSCC1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	YMUL[13:0]													
Initial Value	0	0	0	1	0	0	1	0	1	0	0	1	1	1	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	YMUL[13:0]	H’129F	RW	Y data multiplication coefficient These bits specify a Y data multiplication coefficient in YCbCr → RGB color space conversion. (Initial value: 1.164) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.

35.2.3.82 CRU YCbCr → RGB Color Space Conversion Coefficient 2 Register (ICnCSCC2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	YSUB[11:0]											
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CSUB[11:0]											
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	YSUB[11:0]	H’100	RW	Y data subtraction coefficient These bits specify a Y data subtraction coefficient in YCbCr → RGB color space conversion. Specify an unsigned 12-bit integer. Calculate on 12 bits extended from YCbCr422 8-/10-bit.
15 to 12	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	CSUB[11:0]	H’800	RW	CbCr data subtraction coefficient These bits specify a Cb and Cr data subtraction coefficient in YCbCr → RGB color space conversion. Specify an unsigned 12-bit integer. Calculate on 12 bits extended from YCbCr422 8-/10-bit.

35.2.3.83 CRU YCbCr → RGB Color Space Conversion Coefficient 3 Register (ICnCSCC3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RCRMUL[13:0]													
Initial Value	0	0	0	1	1	0	0	1	1	0	0	0	1	0	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	GCRMUL[13:0]													
Initial Value	0	0	0	0	1	1	0	1	0	0	0	0	0	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 16	RCRMUL [13:0]	H’1989	RW	Cr multiplication coefficient in R data calculation These bits specify a Cr multiplication coefficient for the R data equation in YCbCr → RGB color space conversion. (Initial value: 1.596) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	GCRMUL [13:0]	H’0D02	RW	Cr multiplication coefficient in G data calculation These bits specify a Cr multiplication coefficient for the G data equation in YCbCr → RGB color space conversion. (Initial value: 0.813) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.

35.2.3.84 CRU YCbCr → RGB Color Space Conversion Coefficient 4 Register (ICnCSCC4)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	GCBMUL[13:0]													
Initial Value	0	0	0	0	0	1	1	0	0	1	0	0	0	1	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	BCBMUL[13:0]													
Initial Value	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 16	GCBMUL [13:0]	H’0645	RW	Cb multiplication coefficient in G data calculation These bits specify a Cb multiplication coefficient for the G data equation in YCbCr → RGB color space conversion. (Initial value: 0.392) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.
15, 14	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	BCBMUL [13:0]	H’2045	RW	Cb multiplication coefficient in B data calculation These bits specify a Cb multiplication coefficient for the B data equation in YCbCr → RGB color space conversion. (Initial value: 2.017) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.

35.2.3.85 CRU RGB → YCbCr Color Space Conversion Y Coefficient 1 Register (ICnYCCR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	YCLRP[12:0]												
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	YCLRP[12:0]	H’0107	RW	R multiplication coefficient in Y calculation (R multiplication coefficient for the Y data equation in RGB → YCbCr color space conversion)

35.2.3.86 CRU RGB → YCbCr Color Space Conversion Y Coefficient 2 Register (ICnYCCR2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	YCLBP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	YCLGP[12:0]												
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	YCLBP[12:0]	H’0064	R/W	B multiplication coefficient in Y calculation (B multiplication coefficient for the Y data equation in RGB → YCbCr color space conversion)
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	YCLGP[12:0]	H’0204	R/W	G multiplication coefficient in Y calculation (G multiplication coefficient for the Y data equation in RGB → YCbCr color space conversion)

35.2.3.87 CRU RGB → YCbCr Color Space Conversion Y Coefficient 3 Register (ICnYCCR3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	YCLSFT[4:0]					—	—	—	—	—	—	—	YCLCEN
Initial Value	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	YCLAP[11:0]											
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
30, 29	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 24	YCLSFT[4:0]	H’0A	RW	Amount of downward shift in Y calculation (specification of the amount of downward shift for Y calculation in RGB → YCbCr color space conversion)
23 to 17	—	H’00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	YCLCEN	0b	RW	Y Calculation Data Clip Enable (Data Clipping Processor Control for Y Data Calculation in RGB → YCbCr Color Space Conversion) 0b: Data clipping is not performed, when the Y value is overflow or underflow. 1b: Data clipping is performed Maximum value or Minimum value, when the Y value is overflow or underflow.
15 to 12	—	H’0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	YCLAP[11:0]	H’100	RW	Value added for normalization of Y calculation data (specification of Y data addition constant in RGB → YCbCr color space conversion)

35.2.3.88 CRU RGB → YCbCr Color Space Conversion Cb Coefficient 1 Register (ICnCBCCR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CBCLRP[12:0]												
Initial Value	0	0	0	1	1	1	1	1	0	1	1	0	1	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	CBCLRP [12:0]	H’1F68	RW	R multiplication coefficient in Cb calculation (R multiplication coefficient for the Cb data equation in RGB → YCbCr color space conversion)

35.2.3.89 CRU RGB → YCbCr Color Space Conversion Cb Coefficient 2 Register (ICnCBCCR2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CBCLBP[12:0]												
Initial Value	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CBCLGP[12:0]												
Initial Value	0	0	0	1	1	1	1	0	1	1	0	1	0	1	1	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	CBCLBP [12:0]	H’01C2	RW	B multiplication coefficient in Cb calculation (B multiplication coefficient for the Cb data equation in RGB → YCbCr color space conversion)
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	CBCLGP [12:0]	H’1ED6	RW	G multiplication coefficient in Cb calculation (G multiplication coefficient for the Cb data equation in RGB → YCbCr color space conversion)

35.2.3.90 CRU RGB → YCbCr Color Space Conversion Cb Coefficient 3 Register (ICnCBCCR3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CBCLSFT[4:0]					—	—	—	—	—	—	—	CBCLCEN
Initial Value	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CBCLAP[11:0]											
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
30, 29	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to24	CBCLSFT [4:0]	H'0A	RW	Amount of downward shift in Cb calculation (specification of the amount of downward shift for Cb calculation in RGB → YCbCr color space conversion)
23 to 17	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CBCLCEN	0b	RW	Cb Calculation Data Clip Enable (Data Clipping Process Control for Cb Data Calculation in RGB → YCbCr Color Space Conversion) 0b: Data clipping is not performed. 1b: Data clipping is performed.
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	CBCLAP [11:0]	H'800	RW	Value added for normalization of Cb calculation data (specification of Cb data addition constant in RGB → YCbCr color space conversion)

35.2.3.91 CRU RGB → YCbCr Color Space Conversion Cr Coefficient 1 Register (ICnCRCCR1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CRCLRP[12:0]												
Initial Value	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	CRCLRP [12:0]	H'01C2	RW	R multiplication coefficient in Cr calculation (R multiplication coefficient for the Cr data equation in RGB → YCbCr color space conversion)

35.2.3.92 CRU RGB → YCbCr Color Space Conversion Cr Coefficient 2 Register (ICnCRCCR2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CRCLBP[12:0]												
Initial Value	0	0	0	1	1	1	1	1	1	0	1	1	0	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CRCLGP[12:0]												
Initial Value	0	0	0	1	1	1	1	0	1	0	0	0	0	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 16	CRCLBP [12:0]	H'1FB7	RW	B multiplication coefficient in Cr calculation (B multiplication coefficient for the Cr data equation in RGB → YCbCr color space conversion)
15 to 13	—	000b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
12 to 0	CRCLGP [12:0]	H'1E87	RW	G multiplication coefficient in Cr calculation (G multiplication coefficient for the Cr data equation in RGB → YCbCr color space conversion)

35.2.3.93 CRU RGB → YCbCr Color Space Conversion Cr Coefficient 3 Register (ICnCRCCR3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CRCLSFT[4:0]					—	—	—	—	—	—	—	CRCLCEN
Initial Value	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CRCLAP[11:0]											
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	—	0b	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
30, 29	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28 to 24	CRCLSFT [4:0]	H'0A	RW	Amount of downward shift in Cr calculation (specification of the amount of downward shift for Cr calculation in RGB → YCbCr color space conversion)
23 to 17	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CRCLCEN	0b	RW	Cr Calculation Data Clipping Enable (Data Clipping Process Control for Cr Data Calculation in RGB → YCbCr Color Space Conversion) 0b: Data clipping is not performed. 1b: Data clipping is performed.
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	CRCLAP [11:0]	H'800	RW	Value added for normalization of Cr calculation data (specification of Cr data addition constant in RGB → YCbCr color space conversion)

35.2.3.94 CRU Lookup Table Control Register (ICnLUT)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LUTWR	LUTSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	LUTWR	0b	RW	Enables APB read/write access to the lookup table (LUT). 0b: Disables APB access (set this bit to 0 in operation with the normal path). 1b: Enables APB access (normal access is unavailable in this case). *Note*: **Figure 35.50, Storing LUT Data (Both)** for the LUT setting procedure.
0	LUTSEL	0b	RW	Specifies the access path of the lookup table (LUT). 0b: Normal path (image conversion by using the LUT) 1b: APB I/F (read/write from registers) Used to write (and confirm) data to the LUT in initial setting. Set this bit to 0 after finishing writing and reading data. Note that the LUT data is retained even after a reset. *Note*: **Figure 35.50, Storing LUT Data (Both)** for the LUT setting procedure.

35.2.3.95 CRU Lookup Table Status Register (ICnLUTS)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LUTREGSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	LUTREGSEL	0b	R	Indicates that the lookup table (LUT) can be accessed for a read and write from the APB. The LUTWR register value is reflected in this register after a while. Access the lookup table after confirming this bit. 0b: Both read access and write access are unavailable. 1b: Both read access and write access are available. *Note*: **Figure 35.50, Storing LUT Data (Both)** for the LUT setting procedure.

35.2.3.96 CRU Lookup Table Pointer Register (ICnLUTP)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	LTCRPR[9:0]										LTCBPR[9:6]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTCBPR[5:0]						LTYPR[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 20	LTCRPR[9:0]	H’000	RW	Lookup table (LUT) Cr/B pointer Specify the current pointer (access point) to lookup table Cr/B in APB access. The pointer is incremented automatically when ICnLUTD.LTCRDT[7:0] are written. It returns to 0 upon an overflow. It is not incremented when reading these bits. *Note*: **Figure 35.50, Storing LUT Data (Both)** for the LUT setting procedure.
19 to 10	LTCBPR[9:0]	H’000	RW	Lookup table (LUT) Cb/G pointer Specify the current pointer (access point) to lookup table Cb/G in APB access. The pointer is incremented automatically when ICnLUTD.LTCBDT[7:0] are written. It returns to 0 upon an overflow. It is not incremented when reading these bits. *Note*: **Figure 35.50, Storing LUT Data (Both)** for the LUT setting procedure.
9 to 0	LTYPR[9:0]	H’000	RW	Lookup table (LUT) Y/R pointer Specify the current pointer (access point) to the lookup table Y/R in APB access. The pointer is incremented automatically when ICnLUTD.LUTYDT[7:0] are written. It returns to 0 upon an overflow. It is not incremented when reading these bits. *Note*: **Figure 35.50, Storing LUT Data (Both)** for the LUT setting procedure.

35.2.3.97 CRU Lookup Table Data Register (ICnLUTD)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	LTCRDT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTCBDT[7:0]								LTYDT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	LTCRDT[7:0]	H'00	RW	Lookup table (LUT) Cr/B data Read and write data to the location indicated by the ICnLUTP.LTCRPR[9:0] pointer in lookup table Cr/B through the APB interface.
15 to 8	LTCBDT[7:0]	H'00	RW	Lookup table (LUT) Cb/G data Read and write data to the location indicated by the ICnLUTP.LTCBPR[9:0] pointer in lookup table Cb/G through the APB interface.
7 to 0	LTYDT[7:0]	H'00	RW	Lookup table (LUT) Y/R data Read and write data to the location indicated by the ICnLUTP.LTYPR[9:0] pointer in lookup table Y/R through the APB interface.

35.2.3.98 CRU Test Image Generation Control 1 Register (ICnTICTRL1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	TIRATE[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIPTNV1[3:0]				TIPTNU1[3:0]				TIPTNY1[3:0]				—	—	TIMODE	TIEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	H'000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20 to 16	TIRATE[4:0]	H'00	RW	Specify the test image generation rate. 16 pixels / (18 + TIRATE) cycles
15 to 12	TIPTNV1[3:0]	H'0	RW	Specify the pixel (V) counter bits for the test image pattern. 0000b: Selects bits [3:0]. 0001b: Selects bits [4:1]. 0010b: Selects bits [5:2]. Omitted 1100b: Selects bits [15:12]. 1101b to 1111b: Setting prohibited
11 to 8	TIPTNU1[3:0]	H'0	RW	Specify the pixel (U) counter bits for the test image pattern. 0000b: Selects bits [3:0]. 0001b: Selects bits [4:1]. 0010b: Selects bits [5:2]. Omitted 1100b: Selects bits [15:12]. 1101b to 1111b: Setting prohibited
7 to 4	TIPTNY1[3:0]	H'0	RW	Specify the pixel (Y) counter bits for the test image pattern. 0000b: Selects bits [3:0]. 0001b: Selects bits [4:1]. 0010b: Selects bits [5:2]. Omitted 1100b: Selects bits [15:12]. 1101b to 1111b: Setting prohibited
3, 2	—	00b	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	TIMODE	0b	RW	Specifies a test image generation pattern. 0b: Selects a counter value for YUV output. 1b: Same-color YUV output for the entire screen
0	TIEN	0b	RW	Enables generation of a test image pattern. 0b: Does not generate a test image pattern (processes image data from LINK). 1b: Generates a test image pattern.

35.2.3.99 CRU Test Image Generation Control 2 Register (ICnTICTRL2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TIPTNV2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIPTNU2[7:0]								TIPTNY2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	TIPTNV2[7:0]	H'00	RW	Specify V for the entire-screen same-color test image pattern.
15 to 8	TIPTNU2[7:0]	H'00	RW	Specify U for the entire-screen same-color test image pattern.
7 to 0	TIPTNY2[7:0]	H'00	RW	Specify Y for the entire-screen same-color test image pattern.

35.2.3.100 CRU Test Image Size Setting 1 Register (ICnTISIZE1)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TIPPL[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to12	—	H'0_0000	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	TIPPL[11:0]	H'000	RW	Number of valid pixels per line in the test image pattern *Note 1.* If a value from 0 to 15 is specified, it is assumed that 16 is specified. *Note 2.* The lower four bits are invalid.

35.2.3.101 CRU Test Image Size Setting 2 Register (ICnTISIZE2)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	TIM[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TIN[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	TIM[11:0]	H'000	RW	Invalid period (number of lines) per frame in the test image pattern *Note*: If 0 is specified, it is assumed that 1 is specified.
15 to 12	—	H'0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	TIN[11:0]	H'000	RW	Valid period (number of lines) per frame in the test image pattern *Note*: If 0 is specified, it is assumed that 1 is specified.

35.2.4 Operation

35.2.4.1 Input Formats and Image Processing

Table 35.12 lists the input image formats that can be processed by this module, and corresponding image processing. Output format will be limited depending on this setting.

Table 35.12 Input Format and Image Processing (1/3)

Input Format			Image Processing											Interrupt							Output format					
Data Type	Data Type Classes	Data Type Name	SAV/EAV Decode	Frame Subsampling	Image Clipping	Demosaicing	Data Arrange (Input)	Data Clipping	Linear Matrix	YCbCr to RGB	RGB to YCbCr	LUT	Data Arrange (Output)	YCbCr422 to 420	Frame Start	Frame End	Scan Line	Abnormal Word Count Value	ITU656 DECERR	Statistics	YCbCr422	YCbCr420	RGB	RAW	MIPI CSI-2 Recommended Memory Storage Format	Parallel User-defined binary 16-bit (progressive)
MIPI CSI2 (INF[5:0] Bits of ICnMC Register)																										
H'00	Synchronization Short Packet Data Types	Frame Start Code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'01		Frame End Code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'02		Line Start Code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'03		Line End Code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'04 to H'07		Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'08	Generic Short Packet Data Types	Generic Short Packet Code 1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'09		Generic Short Packet Code 2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'0A		Generic Short Packet Code 3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'0B		Generic Short Packet Code 4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'0C		Generic Short Packet Code 5	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'0D		Generic Short Packet Code 6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'0E		Generic Short Packet Code 7	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'0F		Generic Short Packet Code 8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'10	Generic Long Packet Data Types	Null	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'11		Blanking Data	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'12		Embedded 8-bit non Image Data	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'13		Generic Long Packet Data Type 1	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'14		Generic Long Packet Data Type 2	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'15		Generic Long Packet Data Type 3	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'16		Generic Long Packet Data Type 4	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'17		Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Table 35.12 Input Format and Image Processing (2/3)

Input Format			Image Processing											Interrupt							Output format					
Data Type	Data Type Classes	Data Type Name	SAV/EAV Decode	Frame Subsampling	Image Clipping	Demosaicing	Data Arrange (Input)	Data Clipping	Linear Matrix	YCbCr to RGB	RGB to YCbCr	LUT	Data Arrange (Output)	YCbCr422 to 420	Frame Start	Frame End	Scan Line	Abnormal Word Count Value	ITU656 DECERR	Statistics	YCbCr422	YCbCr420	RGB	RAW	MIPI CSI-2 Recommended Memory Storage Format	Parallel User-defined binary 16-bit (progressive)
H'18	YUV Data	YUV420 8-bit	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'19		YUV420 10-bit	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'1A		Legacy YUV420 8-bit	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'1B		Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'1C		YUV420 8-bit (Chroma Shifted Pixel Sampling)	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'1D		YUV420 10-bit (Chroma Shifted Pixel Sampling)	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'1E		YUV422 8-bit	—	✓	✓	—	✓	✓	—	✓	—	✓	✓	—	✓	✓	✓	✓	—	—	✓	—	✓	—	—	—
H'1F		YUV422 10-bit	—	✓	✓	—	✓	✓	—	✓	—	✓	✓	—	✓	✓	✓	✓	—	—	✓	—	✓	—	—	—
H'20	RGB Data	RGB444	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'21		RGB555	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'22		RGB565	—	✓	✓	—	—	—	—	—	✓	✓	✓	—	✓	✓	✓	✓	—	—	✓	—	✓	—	—	—
H'23		RGB666	—	✓	✓	—	—	—	—	—	✓	✓	✓	—	✓	✓	✓	✓	—	—	✓	—	✓	—	—	—
H'24		RGB888	—	✓	✓	—	—	—	—	—	✓	✓	✓	—	✓	✓	✓	✓	—	—	✓	—	✓	—	—	—
H'25 to H'27		Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
H'28	RAW Data	RAW6	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'29		RAW7	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'2A		RAW8	—	✓	✓	✓	—	—	✓	—	✓	✓	✓	—	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—	—
H'2B		RAW10	—	✓	✓	✓	—	—	✓	—	✓	✓	✓	—	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—	—
H'2C		RAW12	—	✓	✓	✓	—	—	✓	—	✓	✓	✓	—	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—	—
H'2D		RAW14	—	✓	✓	✓	—	—	✓	—	✓	✓	✓	—	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—	—
H'2E		RAW16	—	✓	✓	✓	—	—	✓	—	✓	✓	✓	—	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—	—
H'2F		RAW20	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'30	User Defined Byte-based Data	User Defined 8-bit Data Type 1	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'31		User Defined 8-bit Data Type 2	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'32		User Defined 8-bit Data Type 3	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'33		User Defined 8-bit Data Type 4	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'34		User Defined 8-bit Data Type 5	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'35		User Defined 8-bit Data Type 6	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'36		User Defined 8-bit Data Type 7	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'37		User Defined 8-bit Data Type 8	—	✓	—	—	—	—	—	—	—	—	—	—	✓	✓	—	✓	—	—	—	—	—	—	✓	—
H'38 to H'3F	Reserved	Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Table 35.12 Input Format and Image Processing (3/3)

Input Format			Image Processing											Interrupt							Output format					
Data Type	Data Type Classes	Data Type Name	SAV/EAV Decode	Frame Subsampling	Image Clipping	Demosaicing	Data Arrange (Input)	Data Clipping	Linear Matrix	YCbCr to RGB	RGB to YCbCr	LUT	Data Arrange (Output)	YCbCr422 to 420	Frame Start	Frame End	Scan Line	Abnormal Word Count Value	ITU656 DECERR	Statistics	YCbCr422	YCbCr420	RGB	RAW	MIPI CSI-2 Recommended Memory Storage Format	Parallel User-defined binary 16-bit (progressive)
ITU656/Parallel (PINF[3:0] bit and ITL[1:0] bit of ICnPIFC Register)																										
0000b, 01b	ITU-R BT.656	YCbCr multiplexed 8-bit (Interlace)	✓	✓	—	✓	✓	—	✓	✓	—	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—	—	—
0001b, 01b		YCbCr multiplexed 10-bit (Interlace)*2	✓	✓	—	✓	✓	—	✓	✓	—	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—	—	—
0010b, 00b 0011b, 00b	YCbCr Data	YCbCr parallel 8-bit (Progressive)	—	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	—	✓	✓	✓	—	—	✓	✓	—	✓	—	—	—
0100b, 00b 0101b, 00b 0110b, 00b 0111b, 00b		YCbCr multiplexed 8-bit (Progressive)	—	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	—	✓	✓	✓	—	—	✓	✓	—	✓	—	—	—
1000b, 00b 1001b, 00b 1010b, 00b 1011b, 00b		YCbCr multiplexed 10-bit (Progressive)*2	—	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	—	✓	✓	✓	—	—	✓	✓	—	✓	—	—	—
0010b, 01b 0011b, 01b		YCbCr parallel 8-bit (Interlace)	—	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓	—	—	✓	✓	✓	✓	—	—	—
0100b, 01b 0101b, 01b 0110b, 01b 0111b, 01b		YCbCr multiplexed 8-bit (Interlace)	—	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓	—	—	✓	✓	✓	✓	—	—	—
1000b, 01b 1001b, 01b 1010b, 01b 1011b, 01b		YCbCr multiplexed 10-bit (Interlace)*2	—	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓	—	—	✓	✓	✓	✓	—	—	—
11xxb, 00b	User-defined	Binary 16-bit (Progressive)	—	✓	✓	✓	—	—	✓	—	—	—	—	—	✓	✓	✓	—	—	✓	—	—	—	—	—	✓
Pattern Generator (TIEN Bit of ICnTICTRL1 Register)																										
1b	Test pattern	YCbCr422 8-bit	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	—	—	✓	✓	—	✓	—	—	—

Note 1. Output format

YCbCr422 output format supports YCbCr multiplexed 8bit,YC Separation and Only Y.

YCbCr420 output format supports YCbCr multiplexed 8bit.

RGB output format supports RGB888 (24 bits), RGB888 (32 bits) and ARGB8888 (32 bits).

RAW output format only supports the same width as the input RAW.

Note 2. YUV422 input format

This format is converted from 10 bits to 8 bits by the LUT when ICnMC.LUTTHR = 0, and the upper 8 bits out of 10 bits are converted when ICnMC.LUTTHR = 1.

Note 3. RGB565 and RGB666 input format

This format is once expanded to RGB888 and processed internally. (See **Section 35.2.4.6**)

35.2.4.2 Output Formats

Table 35.13 lists the image output formats.

Table 35.13 Image Output Formats (1/2)

Format			Register	Image Data[255:0]															
				byte31	byte30	byte29	byte28	byte27	byte26	byte25	byte24	byte23	byte22	byte21	byte20	byte19	byte18	byte17	byte16
YUV (YCbCr) 422	Cb 1st		ICnDMR.YCMODE = 001b	Y15 [7:0]	Cr14 [7:0]	Y14 [7:0]	Cb14 [7:0]	Y13 [7:0]	Cr12 [7:0]	Y12 [7:0]	Cb12 [7:0]	Y11 [7:0]	Cr10 [7:0]	Y10 [7:0]	Cb10 [7:0]	Y9 [7:0]	Cr8 [7:0]	Y8 [7:0]	Cb8 [7:0]
	Y 1st		ICnDMR.YCMODE = 000b	Cr14 [7:0]	Y15 [7:0]	Cb14 [7:0]	Y14 [7:0]	Cr12 [7:0]	Y13 [7:0]	Cb12 [7:0]	Y12 [7:0]	Cr10 [7:0]	Y11 [7:0]	Cb10 [7:0]	Y10 [7:0]	Cr8 [7:0]	Y9 [7:0]	Cb8 [7:0]	Y8 [7:0]
	YC Separation		ICnDMR.YCMODE = 010b	Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
				Cr30 [7:0]	Cb30 [7:0]	Cr28 [7:0]	Cb28 [7:0]	Cr26 [7:0]	Cb26 [7:0]	Cr24 [7:0]	Cb24 [7:0]	Cr22 [7:0]	Cb22 [7:0]	Cr20 [7:0]	Cb20 [7:0]	Cr18 [7:0]	Cb18 [7:0]	Cr16 [7:0]	Cb16 [7:0]
	Y component extraction		ICnDMR.YCMODE = 011b	Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
YUV (YCbCr) 420	Cb 1st	Odd	ICnDMR.YCMODE = 101b	Y15 [7:0]	Cr14 [7:0]	Y14 [7:0]	Cb14 [7:0]	Y13 [7:0]	Cr12 [7:0]	Y12 [7:0]	Cb12 [7:0]	Y11 [7:0]	Cr10 [7:0]	Y10 [7:0]	Cb10 [7:0]	Y9 [7:0]	Cr8 [7:0]	Y8 [7:0]	Cb8 [7:0]
		Even		Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
	Y 1st	Odd	ICnDMR.YCMODE = 100b	Cr14 [7:0]	Y15 [7:0]	Cb14 [7:0]	Y14 [7:0]	Cr12 [7:0]	Y13 [7:0]	Cb12 [7:0]	Y12 [7:0]	Cr10 [7:0]	Y11 [7:0]	Cb10 [7:0]	Y10 [7:0]	Cr8 [7:0]	Y9 [7:0]	Cb8 [7:0]	Y8 [7:0]
		Even		Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
RGB-888	24bit/pixel		ICnDMR.RGBMODE = 00b	G10 [7:0]	B10 [7:0]	R9 [7:0]	G9 [7:0]	B9 [7:0]	R8 [7:0]	G8 [7:0]	B8 [7:0]	R7 [7:0]	G7 [7:0]	B7 [7:0]	R6 [7:0]	G6 [7:0]	B6 [7:0]	R5 [7:0]	G5 [7:0]
	32bit/pixel		ICnDMR.RGBMODE = 01b	H'00	R7 [7:0]	G7 [7:0]	B7 [7:0]	H'00	R6 [7:0]	G6 [7:0]	B6 [7:0]	H'00	R5 [7:0]	G5 [7:0]	B5 [7:0]	H'00	R4 [7:0]	G4 [7:0]	B4 [7:0]
ARGB-8888	B 1st		ICnDMR.RGBMODE = 10b	A7 [7:0]	R7 [7:0]	G7 [7:0]	B7 [7:0]	A8 [7:0]	R6 [7:0]	G6 [7:0]	B6 [7:0]	A5 [7:0]	R5 [7:0]	G5 [7:0]	B5 [7:0]	A4 [7:0]	R4 [7:0]	G4 [7:0]	B4 [7:0]
	A 1st		ICnDMR.RGBMODE = 11b	B7 [7:0]	G7 [7:0]	R7 [7:0]	A7 [7:0]	B6 [7:0]	G6 [7:0]	R6 [7:0]	A6 [7:0]	B5 [7:0]	G5 [7:0]	R5 [7:0]	A5 [7:0]	B4 [7:0]	G4 [7:0]	R4 [7:0]	A4 [7:0]
RAW	RAW8		ICnMC.INF=H'2A ICnMC.DEMTHR = 1	P31 [7:0]	P30 [7:0]	P29 [7:0]	P28 [7:0]	P27 [7:0]	P26 [7:0]	P25 [7:0]	P24 [7:0]	P23 [7:0]	P22 [7:0]	P21 [7:0]	P20 [7:0]	P19 [7:0]	P18 [7:0]	P17 [7:0]	P16 [7:0]
	RAW10		ICnMC.INF=H'2B ICnMC.DEMTHR = 1	{H'X, P23[9:0], P22[9:0], P21[9:0], P20[9:0], P19[9:0], P18[9:0]}								{H'X, P17[9:0], P16[9:0], P15[9:0], P14[9:0], P13[9:0], P12[9:0]}							
	RAW12		ICnMC.INF=H'2C ICnMC.DEMTHR = 1	{H'X, P19[11:0], P18[11:0], P17[11:0], P16[11:0], P15[11:0]}								{H'X, P14[11:0], P13[11:0], P12[11:0], P11[11:0], P10[11:0]}							
	RAW14		ICnMC.INF=H'2D ICnMC.DEMTHR = 1	H'X	{P15[13:0], P14[13:0], P13[13:0], P12[13:0]}							H'X	{P11[13:0], P10[13:0], P9[13:0], P8[13:0]}						
	RAW16		ICnMC.INF=H'2E ICnMC.DEMTHR = 1	P14[15:0]		P13[15:0]		P12[15:0]		P11[15:0]		P10[15:0]		P9[15:0]		P8[15:0]		P7[15:0]	
MIPI CSI-2 Recommended Memory Storage Format			ICnMC.CLPTHR = 1 ICnMC.DEMTHR = 1 ICnMC.LMXTHR = 1 ICnMC.CSCTHR = 1 ICnMC.LUTTHR = 1	Byte 32 [7:0]	Byte 31 [7:0]	Byte 30 [7:0]	Byte 29 [7:0]	Byte 28 [7:0]	Byte 27 [7:0]	Byte 26 [7:0]	Byte 25 [7:0]	Byte 24 [7:0]	Byte 23 [7:0]	Byte 22 [7:0]	Byte 21 [7:0]	Byte 20 [7:0]	Byte 19 [7:0]	Byte 18 [7:0]	Byte 17 [7:0]
Parallel User-defined binary 16-bit (progressive)			ICnMC.PINF = H'C - H'F	P15[15:0]		P14[15:0]		P13[15:0]		P12[15:0]		P11[15:0]		P10[15:0]		P9[15:0]		P8[15:0]	

Table 35.13 Image Output Formats (2/2)

Format			Register	Image Data[255:0]															
				byte15	byte14	byte13	byte12	byte11	byte10	byte9	byte8	byte7	byte6	byte5	byte4	byte3	byte2	byte1	byte0
YUV (YCbCr) 422	Cb 1st		ICnDMR.YCMODE = 001b	Y7 [7:0]	Cr6 [7:0]	Y6 [7:0]	Cb6 [7:0]	Y5 [7:0]	Cr4 [7:0]	Y4 [7:0]	Cb4 [7:0]	Y3 [7:0]	Cr2 [7:0]	Y2 [7:0]	Cb2 [7:0]	Y1 [7:0]	Cr0 [7:0]	Y0 [7:0]	Cb0 [7:0]
	Y 1st		ICnDMR.YCMODE = 000b	Cr6 [7:0]	Y7 [7:0]	Cb6 [7:0]	Y6 [7:0]	Cr4 [7:0]	Y5 [7:0]	Cb4 [7:0]	Y4 [7:0]	Cr2 [7:0]	Y3 [7:0]	Cb2 [7:0]	Y2 [7:0]	Cr0 [7:0]	Y1 [7:0]	Cb0 [7:0]	Y0 [7:0]
	YC Separation		ICnDMR.YCMODE = 010b	Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
				Cr14 [7:0]	Cb14 [7:0]	Cr12 [7:0]	Cb12 [7:0]	Cr10 [7:0]	Cb10 [7:0]	Cr8 [7:0]	Cb8 [7:0]	Cr6 [7:0]	Cb6 [7:0]	Cr4 [7:0]	Cb4 [7:0]	Cr2 [7:0]	Cb2 [7:0]	Cr0 [7:0]	Cb0 [7:0]
	Y component extraction		ICnDMR.YCMODE = 011b	Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
YUV (YCbCr) 420	Cb 1st	Odd	ICnDMR.YCMODE = 101b	Y7 [7:0]	Cr6 [7:0]	Y6 [7:0]	Cb6 [7:0]	Y5 [7:0]	Cr4 [7:0]	Y4 [7:0]	Cb4 [7:0]	Y3 [7:0]	Cr2 [7:0]	Y2 [7:0]	Cb2 [7:0]	Y1 [7:0]	Cr0 [7:0]	Y0 [7:0]	Cb0 [7:0]
		Even		Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
	Y 1st	Odd	ICnDMR.YCMODE = 100b	Cr6 [7:0]	Y7 [7:0]	Cb6 [7:0]	Y6 [7:0]	Cr4 [7:0]	Y5 [7:0]	Cb4 [7:0]	Y4 [7:0]	Cr2 [7:0]	Y3 [7:0]	Cb2 [7:0]	Y2 [7:0]	Cr0 [7:0]	Y1 [7:0]	Cb0 [7:0]	Y0 [7:0]
		Even		Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
RGB-888	24bit/pixel		ICnDMR.RGBMODE = 00b	B5 [7:0]	R4 [7:0]	G4 [7:0]	B4 [7:0]	R3 [7:0]	G3 [7:0]	B3 [7:0]	R2 [7:0]	G2 [7:0]	B2 [7:0]	R1 [7:0]	G1 [7:0]	B1 [7:0]	R0 [7:0]	G0 [7:0]	B0 [7:0]
	32bit/pixel		ICnDMR.RGBMODE = 01b	H'00	R3 [7:0]	G3 [7:0]	B3 [7:0]	H'00	R2 [7:0]	G2 [7:0]	B2 [7:0]	H'00	R1 [7:0]	G1 [7:0]	B1 [7:0]	H'00	R0 [7:0]	G0 [7:0]	B0 [7:0]
ARGB-8888	B 1st		ICnDMR.RGBMODE = 10b	A3 [7:0]	R3 [7:0]	G3 [7:0]	B3 [7:0]	A2 [7:0]	R2 [7:0]	G2 [7:0]	B2 [7:0]	A1 [7:0]	R1 [7:0]	G1 [7:0]	B1 [7:0]	A0 [7:0]	R0 [7:0]	G0 [7:0]	B0 [7:0]
	A 1st		ICnDMR.RGBMODE = 11b	B3 [7:0]	G3 [7:0]	R3 [7:0]	A3 [7:0]	B2 [7:0]	G2 [7:0]	R2 [7:0]	A2 [7:0]	B1 [7:0]	G1 [7:0]	R1 [7:0]	A1 [7:0]	B0 [7:0]	G0 [7:0]	R0 [7:0]	A0 [7:0]
RAW	RAW8		ICnMC.INF = H'2A ICnMC.DEMTHR = 1	P15 [7:0]	P14 [7:0]	P13 [7:0]	P12 [7:0]	P11 [7:0]	P10 [7:0]	P9 [7:0]	P8 [7:0]	P7 [7:0]	P6 [7:0]	P5 [7:0]	P4 [7:0]	P3 [7:0]	P2 [7:0]	P1 [7:0]	P0 [7:0]
	RAW10		ICnMC.INF = H'2B ICnMC.DEMTHR = 1	{H'X, P11[9:0], P10[9:0], P9[9:0], P8[9:0], P7[9:0], P6[9:0]}								{H'X, P5[9:0], P4[9:0], P3[9:0], P2[9:0], P1[9:0], P0[9:0]}							
	RAW12		ICnMC.INF = H'2C ICnMC.DEMTHR = 1	{H'X, P9[11:0], P8[11:0], P7[11:0], P6[11:0], P5[11:0]}								{H'X, P4[11:0], P3[11:0], P2[11:0], P1[11:0], P0[11:0]}							
	RAW14		ICnMC.INF = H'2D ICnMC.DEMTHR = 1	H'X	{P7[13:0], P6[13:0], P5[13:0], P4[13:0]}							H'X	{P3[13:0], P2[13:0], P1[13:0], P0[13:0]}						
	RAW16		ICnMC.INF = H'2E ICnMC.DEMTHR = 1	P7[15:0]		P6[15:0]		P5[15:0]		P4[15:0]		P3[15:0]		P2[15:0]		P1[15:0]		P0[15:0]	
MIPI CSI-2 Recommended Memory Storage Format			ICnMC.CLPTHR = 1 ICnMC.DEMTHR = 1 ICnMC.LMXTHR = 1 ICnMC.CSCTHR = 1 ICnMC.LUTTHR = 1	Byte 16 [7:0]	Byte 15 [7:0]	Byte 14 [7:0]	Byte 13 [7:0]	Byte 12 [7:0]	Byte 11 [7:0]	Byte 10 [7:0]	Byte9 [7:0]	Byte8 [7:0]	Byte7 [7:0]	Byte6 [7:0]	Byte5 [7:0]	Byte4 [7:0]	Byte3 [7:0]	Byte2 [7:0]	Byte1 [7:0]
Parallel User-defined binary 16-bit (progressive)			ICnMC.PINF = H'C - H'F	P7[15:0]		P6[15:0]		P5[15:0]		P4[15:0]		P3[15:0]		P2[15:0]		P1[15:0]		P0[15:0]	

Table 35.14 Statistics Output Formats

Format	Data[255:0]															
	byte31	byte30	byte29	byte28	byte27	byte26	byte25	byte24	byte23	byte22	byte21	byte20	byte19	byte18	byte17	byte16
Statistics data	In-block sum of absolute difference data	In-block accumulated data			In-block sum of absolute difference data	In-block accumulated data			In-block sum of absolute difference data	In-block accumulated data			In-block sum of absolute difference data	In-block accumulated data		
	G7 [7:0]	R7 [7:0]	G7 [7:0]	B7 [7:0]	G6 [7:0]	R6 [7:0]	G6 [7:0]	B6 [7:0]	G5 [7:0]	R5 [7:0]	G5 [7:0]	B5 [7:0]	G4 [7:0]	R4 [7:0]	G4 [7:0]	B4 [7:0]

Format	Data[255:0]															
	byte15	byte14	byte13	byte12	byte11	byte10	byte9	byte8	byte7	byte6	byte5	byte4	byte3	byte2	byte1	byte0
Statistics data	In-block sum of absolute difference data	In-block accumulated data			In-block sum of absolute difference data	In-block accumulated data			In-block sum of absolute difference data	In-block accumulated data			In-block sum of absolute difference data	In-block accumulated data		
	G3 [7:0]	R3 [7:0]	G3 [7:0]	B3 [7:0]	G2 [7:0]	R2 [7:0]	G2 [7:0]	B2 [7:0]	G1 [7:0]	R1 [7:0]	G1 [7:0]	B1 [7:0]	G0 [7:0]	R0 [7:0]	G0 [7:0]	B0 [7:0]

35.2.4.3 Transfer to DRAM

(1) Image Data Transfer Destination Address (Image Storage Destination)

- The transfer destination address (image data storage destination) is determined by the AMnMB1ADDRL to AMnMB8ADDRL, and AMnMB1ADDRH to AMnMB8ADDRH registers (MB1ADDR, MB2ADDR, …).
- Up to eight transfer destination addresses (image data storage destinations) are supported.
- A valid register (MB1ADDR, MB2ADDR, …) is switched by AMnMBVALID.
- UV data address offset UVAOF when outputting YUV image data is determined by the AMnUVAOFL and AMnUVAOFH registers.

(a) AMnMBVALID[7:0] = 1111_1111b (eight transfer destinations)

Transfer sequence: MB1 → MB2 → … → MB8 → MB1 → MB2 → …

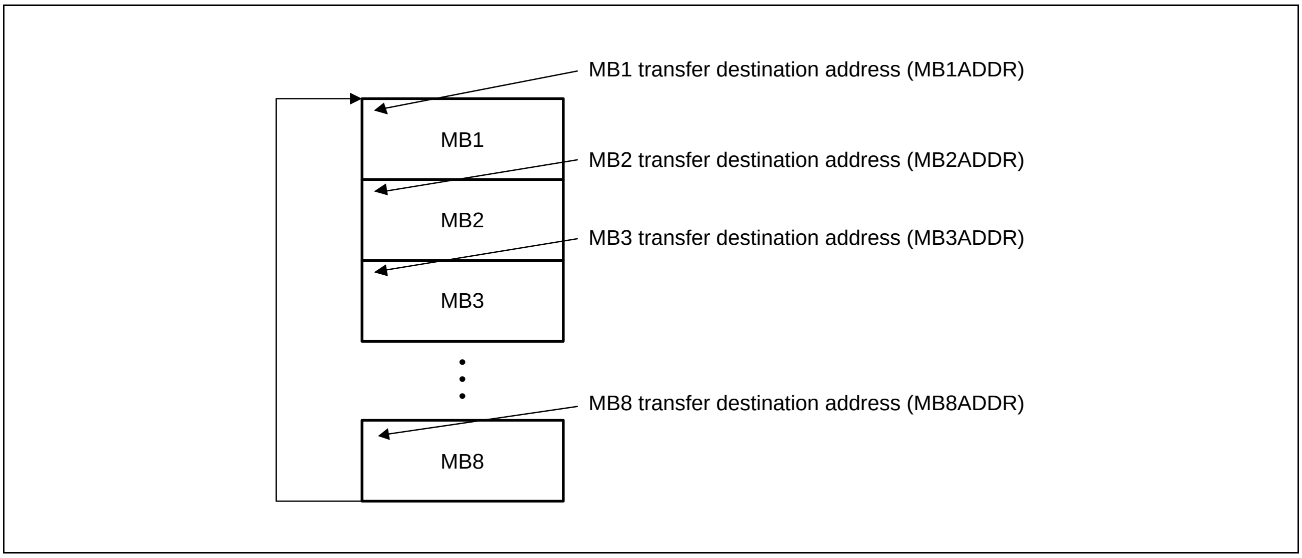


Figure 35.13 Memory Banks (Eight Transfer Destinations)

(b) AMnMBVALID[7:0] = 0000_1111b (four transfer destinations)

MB1 → MB2 → MB3 → MB4 → MB1 → MB2 → …

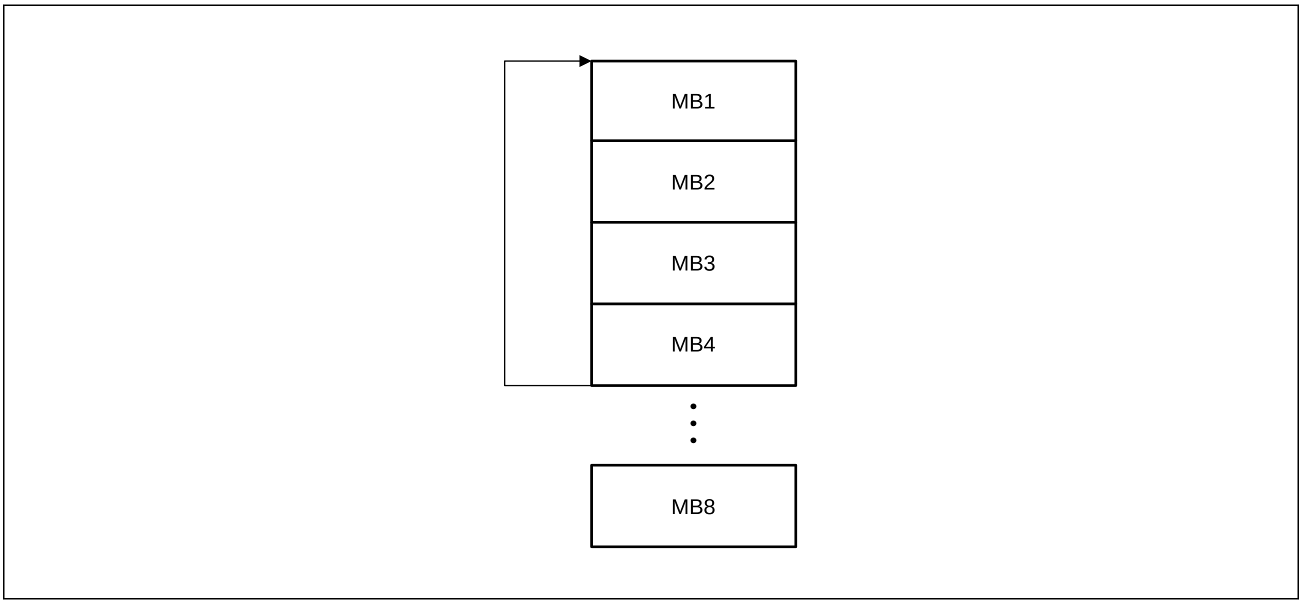


Figure 35.14 Memory Banks (Four Transfer Destinations)

(c) AMnMBVALID[7:0] = 0000_0001b (one transfer destination)

MB1 → MB1 → MB1 → …

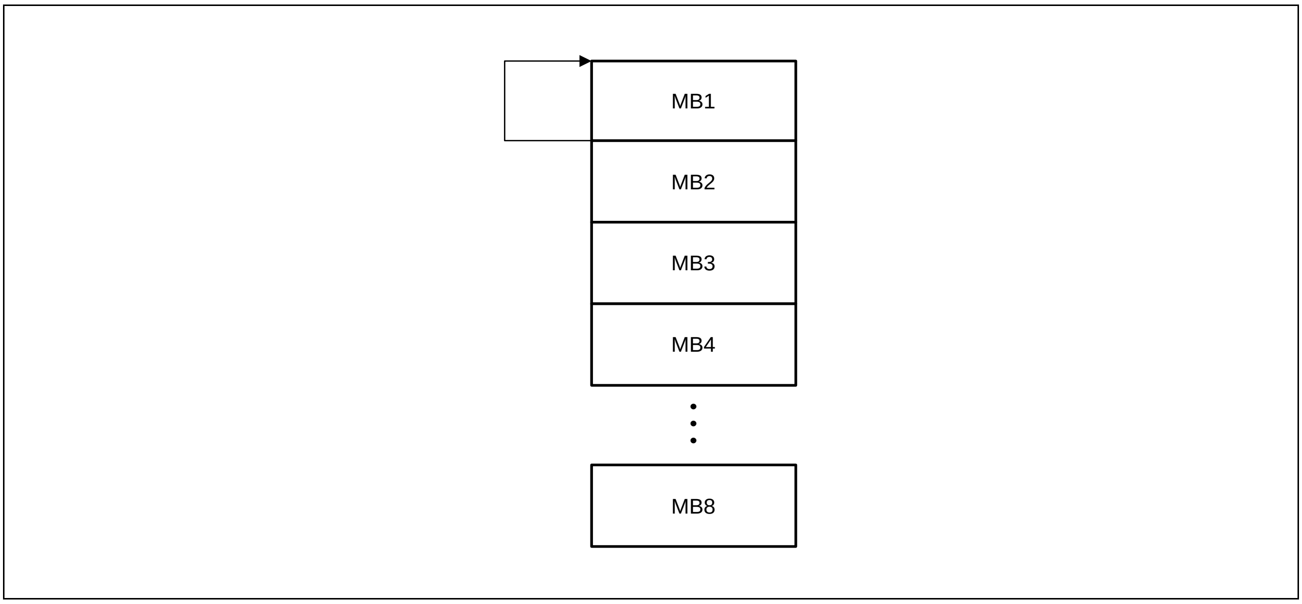


Figure 35.15 Memory Banks (One Transfer Destinations)

(2) Image Data Transfer Destination Address (Storage in Y/UV Separation)

The following shows the sequence of data storage in the MB1 transfer destination address (MB1ADDR) in the case of Y First (YUYV...).

(a) When YUV is Not Separated (ICnDMR.YCMODE[2:0] = 000b to 001b, 100b to 101b):

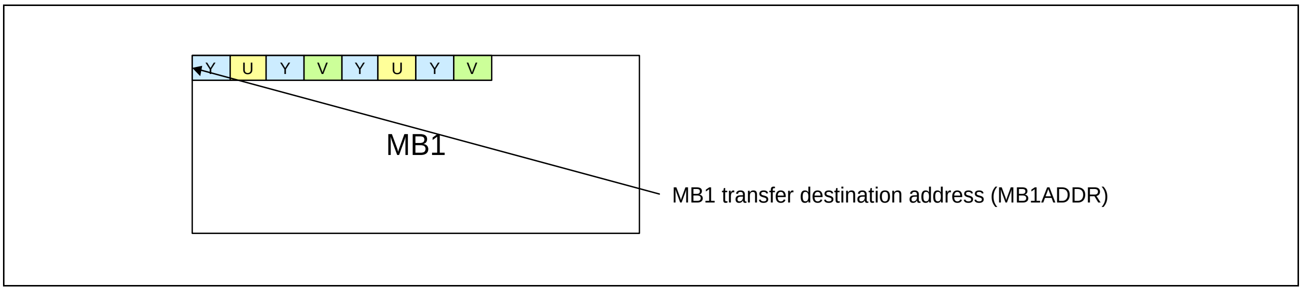


Figure 35.16 Data Storage (YUV is Not Separated)

(b) When YUV is Separated (ICnDMR.YCMODE[2:0] = 010b):

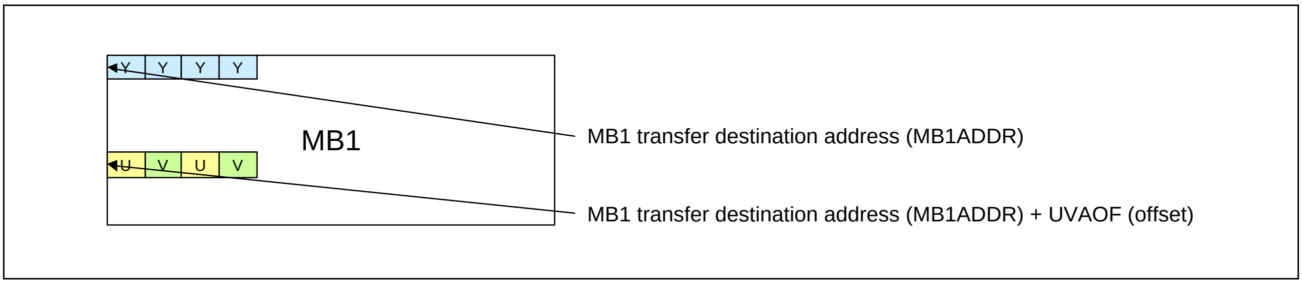


Figure 35.17 Data Storage (YUV is Separated)

(c) When Only Y is Separated (ICnDMR.YCMODE[2:0] = 011b)

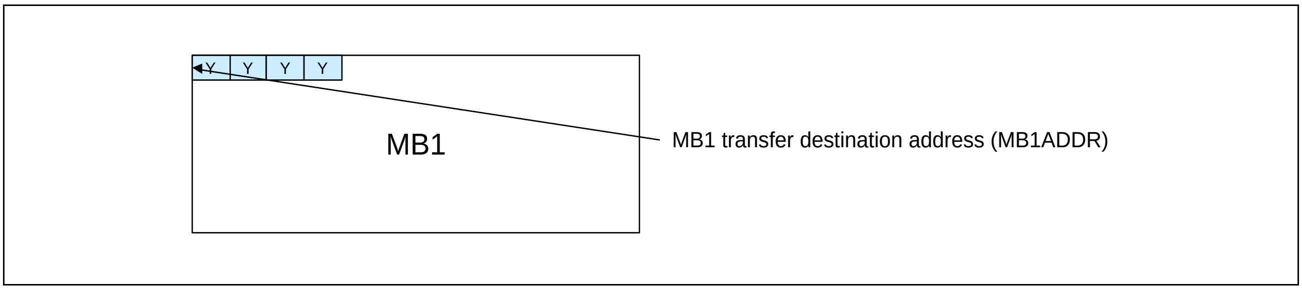


Figure 35.18 Data Storage (Only Y is Separated)

(3) Image Data Transfer Destination Address (Storage in Interlaced Mode)

The following shows the data storage sequence for interlaced data that the parallel input supports.

(a) When YUV is Not Separated (ICnDMR.YCMODE[2:0] = 000b to 001b, 100b to 101b):

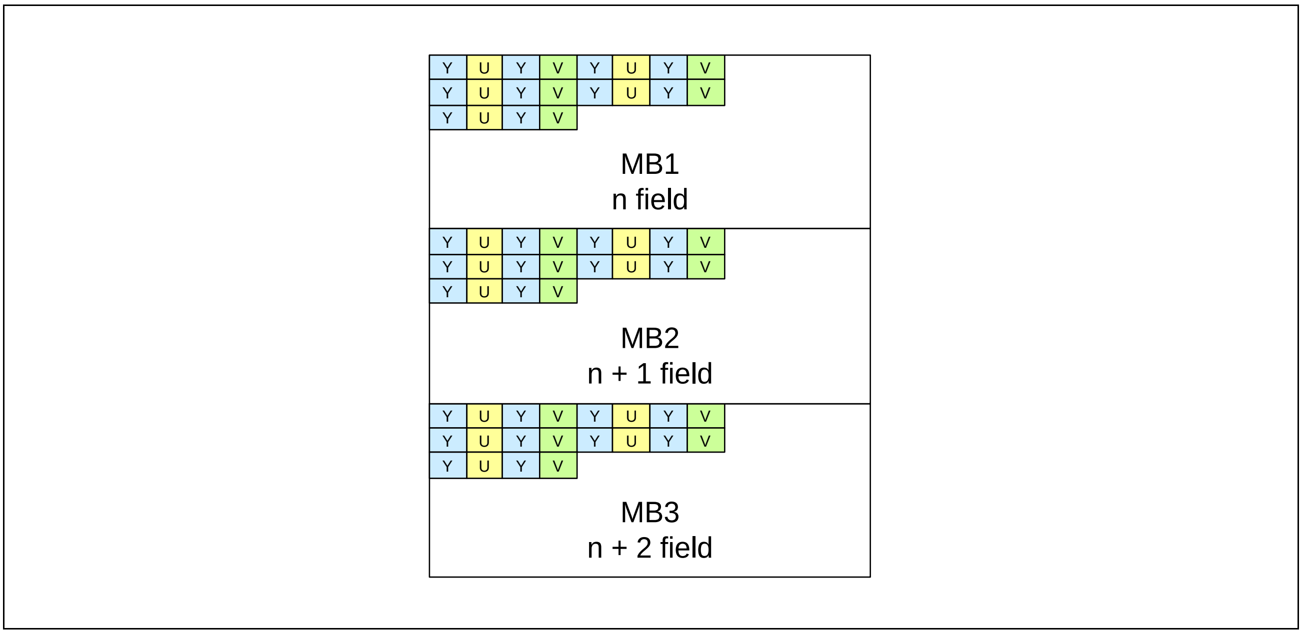


Figure 35.19 Data Storage (Interlaced, YUV is Not Separated)

(b) When YUV is Separated (ICnDMR.YCMODE[2:0] = 010b):

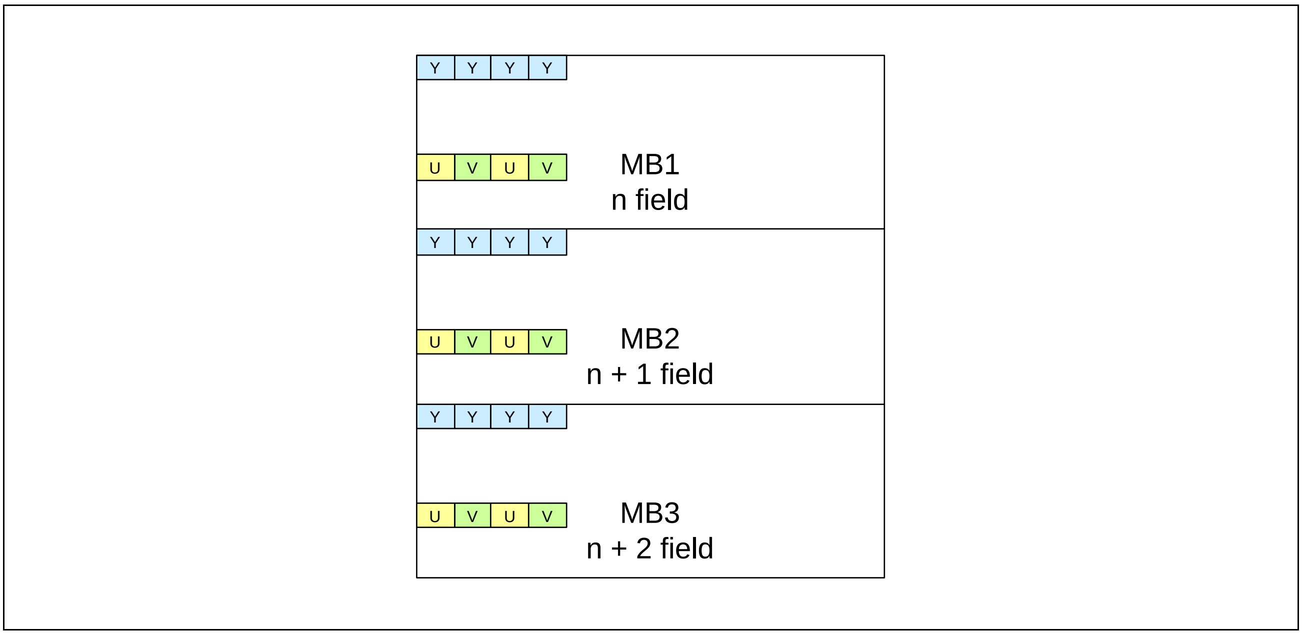


Figure 35.20 Data Storage (Interlaced, YUV is Separated)

(c) When Only Y is Separated (ICnDMR.YCMODE[2:0] = 011b)

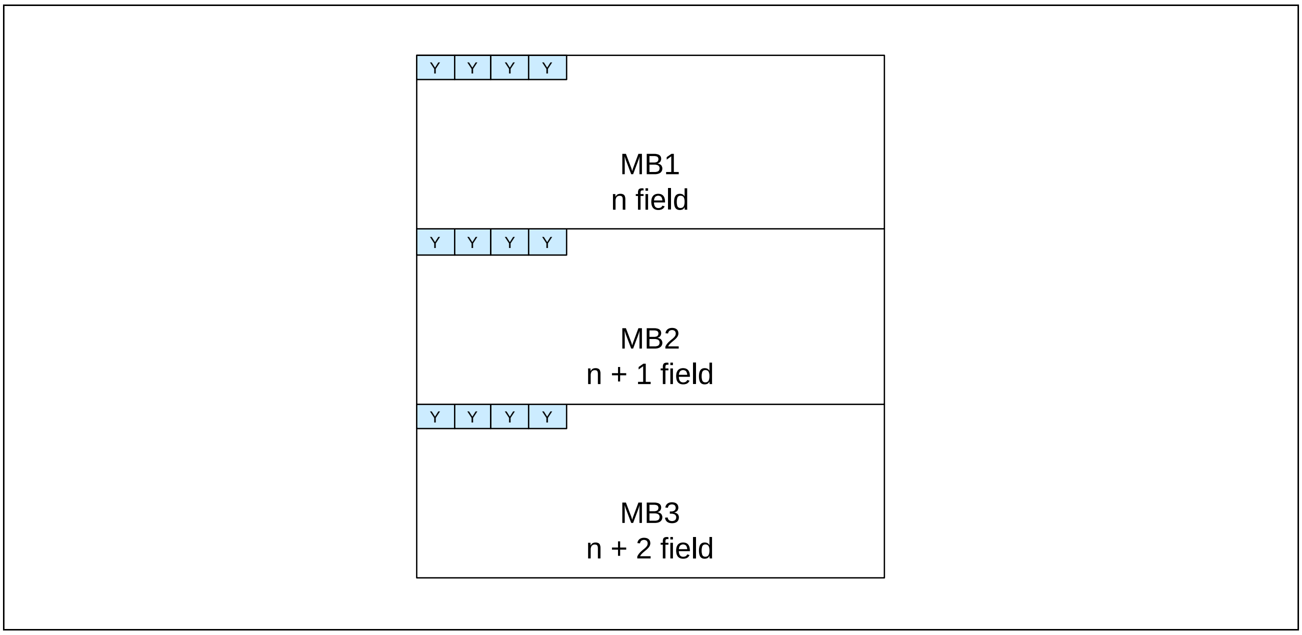


Figure 35.21 Data Storage (Interlaced, Only Y is Separated)

(d) Memory Bank Write Order by CAM_FIELD Pin Input and ICnPIFC.ITL[2] Register Setting

The order of writing to the memory bank changes depending on the CAM_FIELD input and the ICnPIFC.ITL[2] register setting. See **Table 35.15** for details.

Table 35.15 Relationship between CAM_FIELD/ICnPIFC.ITL[2] and Memory Bank Write Order

CAM_FIELD	ICnPIFC.ITL[2]	Write Order to Memory Banks
Start from 0 0 → 1 → 0 → 1 → 0 →	0	Writes to memory banks in ascending order. e.g. when MBVALID[7:0] = H'FF is setting MB1 → MB2 → MB3 → MB4 → MB5 → MB6 → MB7 → MB8 →
	1	Even and odd numbers are replaced and written to the memory bank e.g. when MBVALID[7:0] = H'FF is setting MB2 → MB1 → MB4 → MB3 → MB6 → MB5 → MB8 → MB7 →
Start from 1 1 → 0 → 1 → 0 → 1 →	0	Even and odd numbers are replaced and written to the memory bank e.g. when MBVALID[7:0] = H'FF is setting MB2 → MB1 → MB4 → MB3 → MB6 → MB5 → MB8 → MB7 →
	1	Writes to memory banks in ascending order. e.g. when MBVALID[7:0] = H'FF is setting MB1 → MB2 → MB3 → MB4 → MB5 → MB6 → MB7 → MB8 →

(4) Statistics Data Transfer Destination Address (Statistics Data Storage Destination)

The results of statistics processing are transferred from the statistics data bus to external memory. The external transfer destination start address is determined by AMnSDMB1 to 8ADDRL and AMnSDMB1 to 8ADDRH.

The number of planes of the memory bank is determined by AMnSDMBVALID. The transfer start address (statistics data storage destination) is determined by the MnSDMB1 to 8ADDRL and AMnSDMB1 to 8ADDRH registers.

The valid AMnSDMB1ADDR, AMnSDMB2ADDR, …registers are switched by AMnSDMBVALID.

(a) AMnSDMBVALID[7:0] = 1111_1111b (eight transfer destinations)

Order of transfer destinations: SDMB1 → SDMB2 → … → SDMB8 → SDMB1 → SDMB2 → …

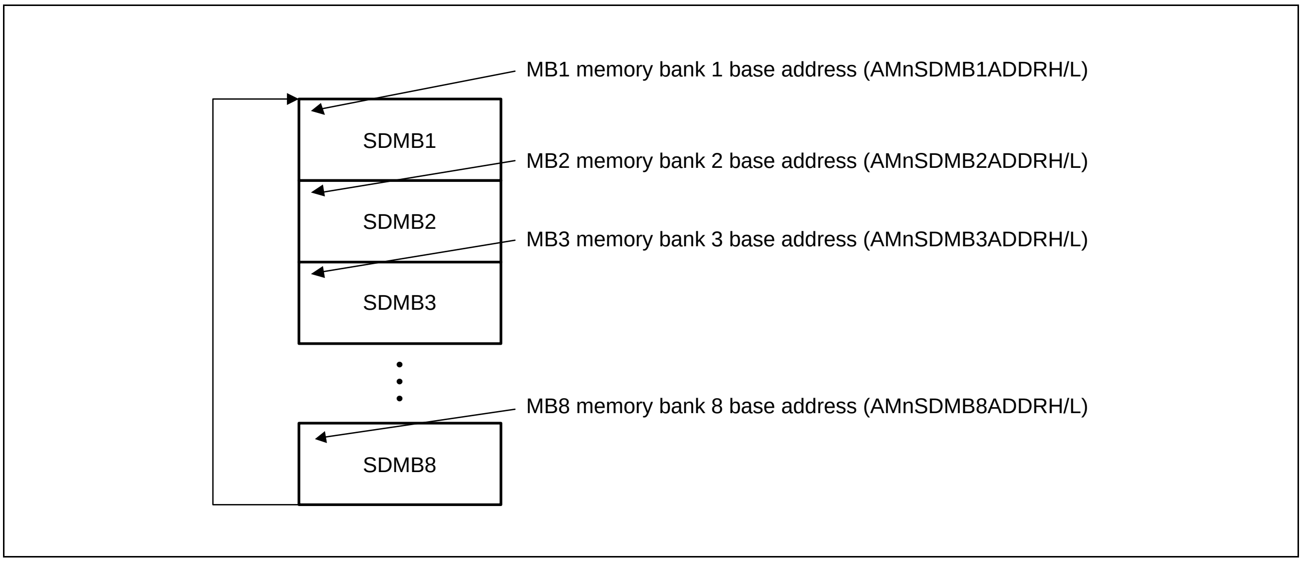


Figure 35.22 Image of Memory Bank (Eight Transfer Destinations)

(5) Processing of the Last Line in a Frame

Note that data is written to the end of the frame (32 bytes × number of AXI bursts specified by the AXILEN[3:0] bits in the AMnAXIATTR register). Because overrun data is written for a frame, ignore it. Set the start address taking this into consideration.

The figure below shows the case of full HD YUV422.

Figure 35.23 Data Storage

35.2.4.4 SAV/EAV Decode

This function relates to the parallel input.

The function detects the data start and data end as defined by the ITU-R BT.656 Standard. If the preamble data (SAV/EAV) is invalid, 1-bit error data can be corrected by the error correction function (switched by the EC bit of the ICnPIFC register) if enabled. If 2-bit error data is received, it is not corrected. In this case, an interrupt signal (image_conv_err_int) that indicates that an invalid format has been received is asserted. (CRUnIE.CEE must be set to 1.) This function continues its operation from the next line even if an error occurs.

35.2.4.5 Parallel User-Defined Binary 16-Bit (Progressive)

User-defined binary 16-bit (progressive) from the parallel interface can be output to memory as user-defined binary 16-bit (progressive) in that format.

This mode can be supported by setting the register as follows.

CRUnCTRL.VINSEL = 1
ICnMC.PINF = H’C, H’D, H’E or H’F

35.2.4.6 RGB565/RGB666 to RGB888

RGB565 and RGB666 inputs are converted to RGB888 as shown in **Table 35.16**.

Table 35.16 Conversion Table from RGB565/RGB666 to RGB888

RGB565	RGB888
R[4:0]	{R[4:0], R[4:2]}
G[5:0]	{G[5:0], G[5:4]}
B[4:0]	{B[4:0], B[4:2]}

RGB666	RGB888
R[5:0]	{R[5:0], R[5:4]}
G[5:0]	{G[5:0], G[5:4]}
B[5:0]	{B[5:0], B[5:4]}

35.2.4.7 MIPI CSI-2 Recommended Storage Format

In the output format, there are data types that can be output in MIPI CSI-2 Recommended Memory Storage Format.

This format packs the received data in byte units and outputs them.

The data is packed according to the method specified for each Data Type in the MIPI CSI-2 standard, so please refer to the standard for details such as bit positions.

(1) Mode Setting

This mode can be supported by setting the ICnMC register as follows.
ICnMC.LUTTHR = 1
ICnMC.DEMTHR = 1
ICnMC.LMXTHR = 1
ICnMC.CSCTHR = 1
ICnMC.CLPTHR = 1

(2) Supported Data Types

See to **Table 35.12** for supported datatypes.

35.2.4.8 Image Clipping

This function clips the input image to the range specified by registers. Specify the clipping range by setting the following registers.

(Horizontal direction)

- Image clipping start pixel register: ICnSPPrC.SPPrC[11:0]
- Image clipping end pixel register: ICnEPPrC.EPPrC[11:0]

(Vertical direction)

- Image clipping start line register: ICnSLPrC.SLPrC[11:0]
- Image clipping end line register: ICnELPrC.ELPrC[11:0]

While the image clipped to the size which is indicated in grey is being processed, the AV bit of the ICnMS register is set to 1. If any other part of the image is being processed, the AV bit of the ICnMS register is set to 0. An image is clipped in the same way even if the image is subsampled.

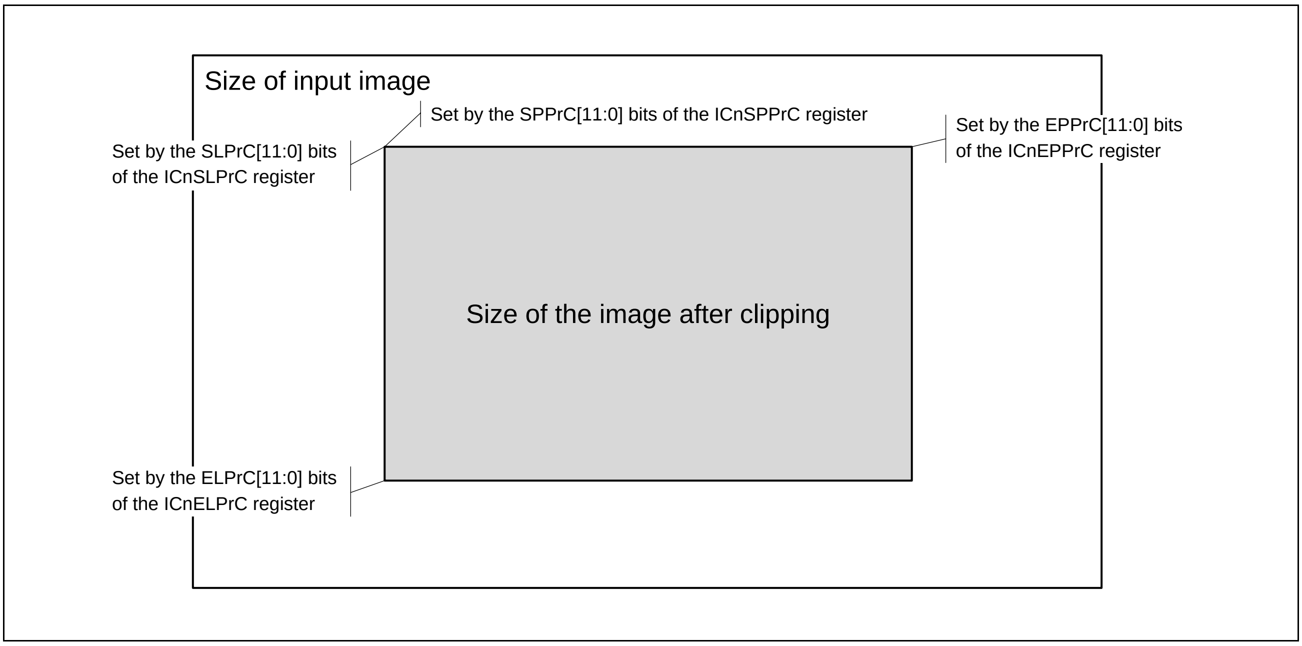


Figure 35.24 Image Clipping

35.2.4.9 Frame Subsampling

This function subsamples the input image by setting a register. Set the following register to specify the reduced frequency of transmission (once every × frames).

- Frame subsampling control register: ICnDEC.FRMDEC[3:0]

The following shows the operation when ICnDEC.FRMDEC[3:0] is set to 2.

The image is sampled every (register value + 1) frames. ICnMS.CA is set to 1 when a non-subsampled image is being processed.

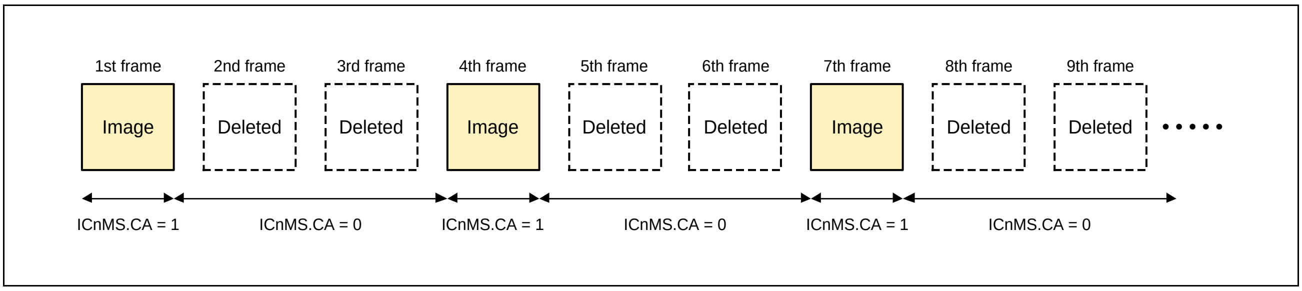


Figure 35.25 Frame Subsampling When ICnDEC.FRMDEC[3:0] is Set to 2 (Progressive)

In interlaced mode, the top field and bottom field are handled as one set (i.e., processed every two frames).

For example, when ICnDEC.FRMDEC[3:0] is set to 2, the first and second frames are output, the third to sixth frames are ignored, and then the seventh and eighth frames are output.

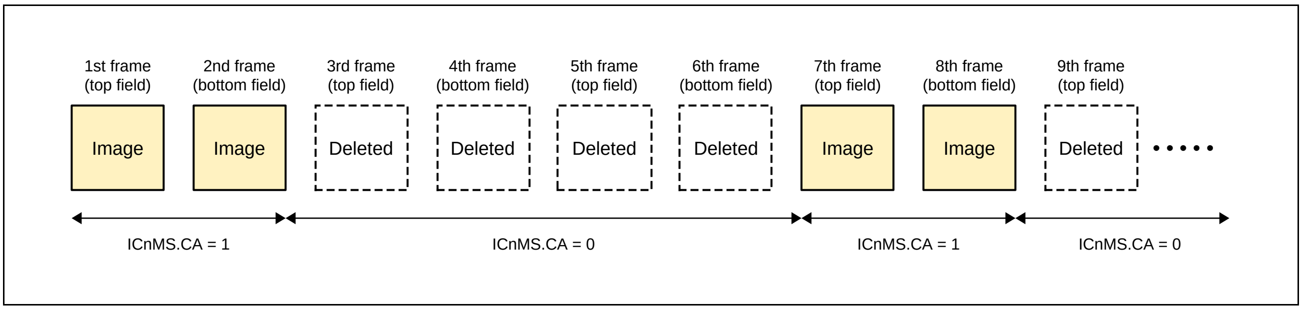


Figure 35.26 Frame Subsampling When ICnDEC.FRMDEC[3:0] is Set to 2 (Interlaced)

35.2.4.10 Demosaicing

This function is only valid on input data of the RAW type. Its operation is on the assumption that the RAW data input from MIPI-CSI-2 are Bayer data and generates pixel data that are interpolated from the RAW data.

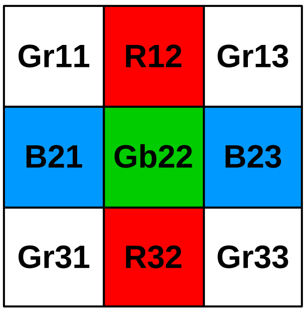
The demosaicing function uses a bilinear method. The 2×2 pixels surrounding the calculation target position are used to linearly interpolate the luminance. The interpolation formula is shown below. The edges of the image are processed by mirroring as follows.

The following describes the method of calculating the RGB components (R22, Gb22/Gr22, B22) of the pixels at the center position.

In the case of RAW10/12, the eight higher-order bits are used for the calculation.

(1) Calculating the Gb22 data portion

Figure 35.27 shows the method of interpolation calculation for pixel data (R/G/B) of Gb22.



$$R22 = (R12 + R32)/2$$

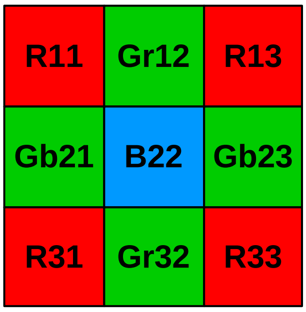
$$Gb22 = Gb22$$

$$B22 = (B21 + B23)/2$$

Figure 35.27 Calculation of the Gb22 Data

(2) Calculating the B22 data portion

Figure 35.28 shows the method of interpolation calculation for pixel data (R/G/B) of B22.



$$R22 = (R11 + R13 + R31 + R33)/4$$

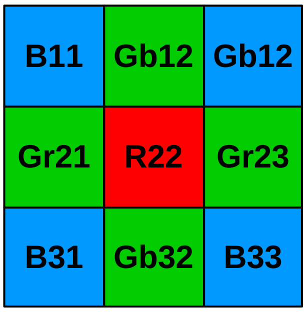
$$Gb22 = (Gr12 + Gb21 + Gb23 + Gr32)/4$$

$$B22 = B22$$

Figure 35.28 Calculation of the B22 Data Portion

(3) Calculating the R22 data portion

Figure 35.29 shows the method of interpolation calculation for pixel data (R/G/B) of R22.



B11	Gb12	Gb12
Gr21	R22	Gr23
B31	Gb32	B33

R22 = R22

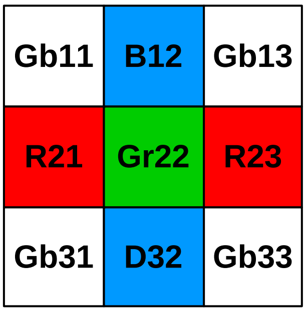
Gr22 = (Gb12 + Gr21 + Gr23 + Gb32)/4

B22 = (B11 + B13 + B31 + B33)/4

Figure 35.29 Calculation of the R22 Data Portion

(4) Calculating the Gr22 data portion

Figure 35.30 shows the method of interpolation calculation for pixel data (R/G/B) of G22.



Gb11	B12	Gb13
R21	Gr22	R23
Gb31	D32	Gb33

R22 = (R21 + R23)/2

Gr22 = Gr22

B22 = (B12 + B32)/2

Figure 35.30 Calculation of the Gr22 Data Portion

35.2.4.11 Linear Matrix

Linear matrix processing involves calculation of the matrix of R’, G’, and B’ (8 bits per color) by using the registers listed below to set the following coefficients for R, G, and B (8 bits per color) after demosaicing as the previous stage.

Gof = Register ICnLMXOF.GOF[7:0]
Rof = Register ICnLMXOF.ROF[7:0]
Bof = Register ICnLMXOF.BOF[7:0]

Rr = Register ICnLMXRC1.RR[12:0]
Rb = Register ICnLMXRC2.RB[12:0]
Rg = Register ICnLMXRC2.RG[12:0]

Gr = Register ICnLMXGC1.GR[12:0]
Gb = Register ICnLMXGC2.GB[12:0]
Gg = Register ICnLMXGC2.GG[12:0]

Br = Register ICnLMXBC1.BR[12:0]
Bb = Register ICnLMXBC2.BB[12:0]
Bg = Register ICnLMXBC2.BG[12:0]

$$\mathrm{R'} = \frac{\mathrm{Rr} \cdot (\mathrm{R} + \mathrm{Rof})^{*1} + \mathrm{Rg} \cdot (\mathrm{G} + \mathrm{Gof})^{*1} + \mathrm{Rb} \cdot (\mathrm{B} + \mathrm{Bof})^{*1}}{1024}$$

$$\mathrm{G'} = \frac{\mathrm{Gr} \cdot (\mathrm{R} + \mathrm{Rof})^{*1} + \mathrm{Gg} \cdot (\mathrm{G} + \mathrm{Gof})^{*1} + \mathrm{Gb} \cdot (\mathrm{B} + \mathrm{Bof})^{*1}}{1024}$$

$$\mathrm{B'} = \frac{\mathrm{Br} \cdot (\mathrm{R} + \mathrm{Rof})^{*1} + \mathrm{Bg} \cdot (\mathrm{G} + \mathrm{Gof})^{*1} + \mathrm{Bb} \cdot (\mathrm{B} + \mathrm{Bof})^{*1}}{1024}$$

Note 1. When the result of addition is a negative value, it is converted to 0.

35.2.4.12 Data Arrange

This function converts the negative expression of UV in the YUV (YCbC) data when inputting and outputting image data. The Data Arrange1 input side is set by the IBINSEL bit of the ICnMC register, and the Data Arrange2 output side is set by the OBINSEL bit of the ICnDMR register.

Set Data Arrange1 according to the sensor specifications. Data Arrange1 supports 8 or 10 bits depending on the number of bits of input data. Set Data Arrange2 according to the specifications used by the output destination. Data Arrange2 supports 8 bits depending on the output data. YUV of the MIPI CSI-2 module is usually used in offset binary. It may be output in two’s complement depending on the sensor used.

Table 35.17 shows an example of processing when the input is 8 bits.

Data Arrange1 also has 10-bit input, and when 10-bit input is used, it can be converted to 10-bit offset binary/2’s complement as well as 8-bit.

Table 35.17 Input Format and Image Processing

UV Value (Hex)	Offset Binary (Dec) Register Settings: DataArrange1: ICnMC.IBINSEL = 0 DataArrange2: ICnDMR. OBINSEL = 0	Two’s Complement (Dec) Register Settings: DataArrange1: ICnMC.IBINSEL = 1 DataArrange2: ICnDMR. OBINSEL = 1
H’FF	127	−1
H’FE	126	−2
. . .	. . .	…
H’81	1	−127
H’80	0	−128
H’7F	−1	127
. . .	…	…
H’01	−127	1
H’00	−128	0

35.2.4.13 Data Clipping

Data in the YCbCr (YUV) format is clipped. To make the input data to be RGB-converted comply with the ITU-R BT.601 standard, specify whether to clip the values below 16 and above 240 for YCbCr-8bit and below 64 and above 960 for YCbCr-10bit. Data clipping can be set by the CLP[1:0] bits in the ICnMC register. (See **Section 35.2.3.58**.)

35.2.4.14 Color Space Conversion

(1) RGB to YCbCr

The RGB data (8 or 10 bits) which is image-clipped, frame-subsampled, or demosaiced is extended to 12 bits, and then is converted into YCbCr (10 bits). By setting coefficients in registers, perform calculations equivalent to ITU-R BT.601 and ITU-R BT.709.

Perform calculations according to the coefficients and formulas specified by the following registers:

YCLRP = ICnYCCR1.YCLRP[12:0]
YCLBP = ICnYCCR2.YCLBP[12:0]
YCLGP = ICnYCCR2.YCLGP[12:0]
YCLSFT = ICnYCCR3.YCLSFT[4:0]
YCLAP = ICnYCCR3.YCLAP[11:0]

CBCLRP = ICnCBCCR1.CBCLRP[12:0]
CBCLBP = ICnCBCCR2.CBCLBP[12:0]
CBCLGP = ICnCBCCR2.CBCLGP[12:0]
CBCLSFT = ICnCBCCR3.CBCLSFT[4:0]
CBCLAP = ICnCBCCR3.CBCLAP[11:0]

CRCLRP = ICnCRCCR1.CRCLRP[12:0]
CRCLBP = ICnCRCCR2.CRCLBP[12:0]
CRCLGP = ICnCRCCR2.CRCLGP[12:0]
CRCLSFT = ICnCRCCR3.CRCLSFT[4:0]
CRCLAP = ICnCRCCR3.CRCLAP[11:0]

$Y = ((\mathrm{YCLRP} \times R + \mathrm{YCLGP} \times G + \mathrm{YCLBP} \times B) / 2^{\mathrm{YCLSFT}}) + \mathrm{YCLAP}$
$Cb = ((\mathrm{CBCLRP} \times R + \mathrm{CBCLGP} \times G + \mathrm{CBCLBP} \times B) / 2^{\mathrm{CBCLSFT}}) + \mathrm{CBCLAP}$
$Cr = ((\mathrm{CRCLRP} \times R + \mathrm{CRCLGP} \times G + \mathrm{CRCLBP} \times B) / 2^{\mathrm{CRCLSFT}}) + \mathrm{CRCLAP}$

In the case of ITU-R BT.601 (8 bits) (same as defaults):

$Y = 0.257 \times R + 0.504 \times G + 0.098 \times B + 16$
$Cb = -0.148 \times R - 0.291 \times G + 0.439 \times B + 128$
$Cr = 0.439 \times R - 0.368 \times G - 0.071 \times B + 128$

Register Settings:

YCLRP = H’0107
YCLBP = H’0064
YCLGP = H’0204
YCLSFT = H’0A
YCLAP = H’100

CBCLRP = H’1F68
CBCLBP = H’01C2
CBCLGP = H’1ED6
CBCLSFT = H’0A
CBCLAP = H’800

CRCLRP = H’01C2
CRCLBP = H’1FB7
CRCLGP = H’1E87
CRCLSFT = H’0A
CRCLAP = H’800

(2) YCbCr to RGB

The YCbCr data (8 or 10 bits) which is data-arranged and data-clipped is extended to 12 bits (the extension method: 0 added to lower bits), and then is converted into RGB (10 bits). By setting coefficients in registers, perform calculations equivalent to ITU-R BT.601 and ITU-R BT.709. Perform calculations according to the coefficients and formulas specified by the following registers:

YMUL = ICnCSCC1.YMUL[13:0]
YSUB = ICnCSCC2.YSUB[11:0]
RCRMUL = ICnCSCC2.CSUB[11:0]
GCRMUL = ICnCSCC3.RCRMUL[13:0]
GCBMUL = ICnCSCC3.GCRMUL[13:0]
BCBMUL = ICnCSCC4.GCBMUL[13:0]
CSUB = ICnCSCC4.BCBMUL[13:0]

$$\mathrm{R} = \frac{\mathrm{YMUL} \times (\mathrm{Y} - \mathrm{YSUB})^{*1} + \mathrm{RCRMUL} \times (\mathrm{Cr} - \mathrm{CSUB})}{4096}$$

$$\mathrm{G} = \frac{\mathrm{YMUL} \times (\mathrm{Y} - \mathrm{YSUB})^{*1} - \mathrm{GCRMUL} \times (\mathrm{Cr} - \mathrm{CSUB}) - \mathrm{GCBMUL} \times (\mathrm{Cb} - \mathrm{CSUB})}{4096}$$

$$\mathrm{B} = \frac{\mathrm{YMUL} \times (\mathrm{Y} - \mathrm{YSUB})^{*1} + \mathrm{BCBMUL} \times (\mathrm{Cb} - \mathrm{CSUB})}{4096}$$

Note 1. A negative value is rounded to 0.

In the case of ITU-R BT.601 (8 bits) (same as defaults):

$\mathrm{R} = 1.164 \times (\mathrm{Y} - 16) + 1.596 \times (\mathrm{Cr} - 128)$
$\mathrm{G} = 1.164 \times (\mathrm{Y} - 16) - 0.813 \times (\mathrm{Cr} - 128) - 0.392 \times (\mathrm{Cb} - 128)$
$\mathrm{B} = 1.164 \times (\mathrm{Y} - 16) + 2.017 \times (\mathrm{Cb} - 128)$

Register Settings:

YMUL = H’129F
YSUB = H’100
RCRMUL = H’1989
GCRMUL = H’0D02
GCBMUL = H’0645
BCBMUL = H’2045
CSUB = H’800

35.2.4.15 LUT (Lookup Table)

Used to convert 10-bit data after color space conversion (YCbCr to RGB or RGB to YCbCr) into 8-bit data.

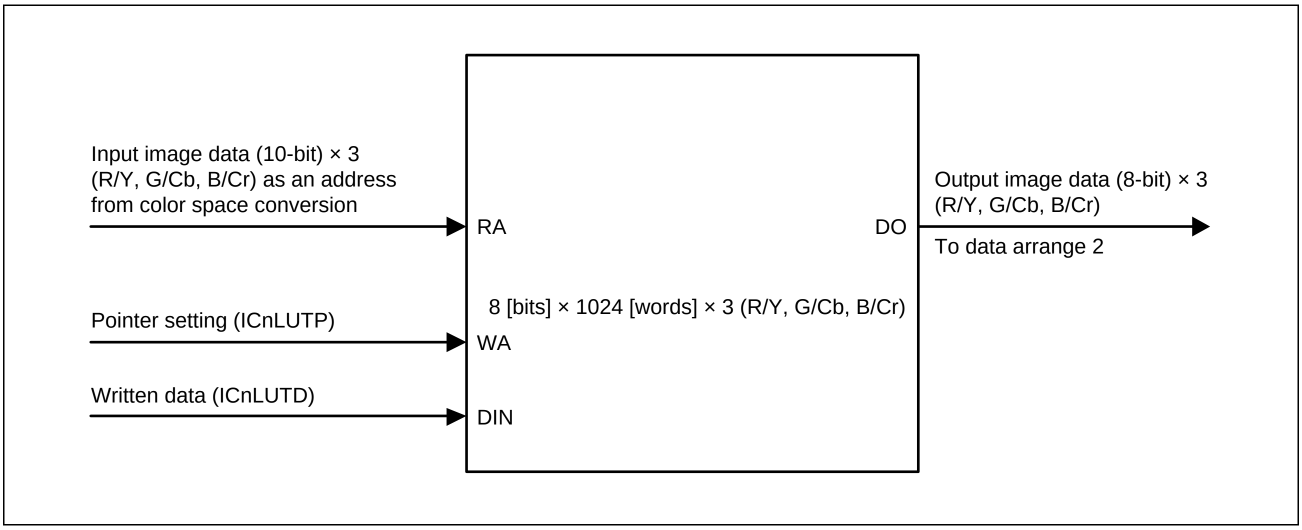


Figure 35.31 Lookup Table (LUT)

(1) Pointer Access

The pointer is incremented every time the ICnLUTD register is written.

The incremented value can be read from the ICnLUTP register. When the ICnLUTD register is read, the pointer is not incremented. The pointer must be set immediately before the register is read.

Follow the procedure below to access the lookup table. See **Section 35.3.1**.

The lookup table cannot be accessed from the CPU in normal mode.

(2) Lookup Table Usage Notes

1. Set the lookup table only after the CRU has stopped its operation.
2. Rewrite the entire lookup table before use because the initial values are not guaranteed after the power is turned on.
3. Data to be set in the lookup table must be restricted to the range that is compatible with the input bit width.

35.2.4.16 YCbCr422 to 420

Data is skipped from YCbCR422 to produce YCbCr420. Averaging or any other calculations are not performed. This conversion can be set by the YCMODE[2:0] bits in the ICnDMR register. (See **Section 35.2.3.80**.)

35.2.4.17 Y/C Separation / Only Y

Only Y can be separated or extracted from Y/C, and then can be stored in memory. For details, see **Section 35.2.4.3(2)**.

35.2.4.18 Statistics

This function is limited to a RAW data input through SVC0 of CRU0/1. This module divides one image vertically and horizontally into blocks (see **Figure 35.32**). The statistical information within this block is transferred to memory and the image is adjusted by weighting the information of every block by the CPU and providing feedback to the external video state.

1) In-block accumulation

Accumulation of each color component, R, G, and B, is performed within each block.

On the CPU side, the weighting is applied per block based on the information from each block, and the obtained evaluation values can be used for exposure control, etc. from the next frame.

The weighting could in general be center-weighted allocation, weighted allocation according to the subject, and so on. If the subject is judged to be dark, exposure control is applied to widen the aperture, and if the subject is judged to be bright, the aperture is made narrower.

Additionally, each color component among the R, G, and B components is adjusted and the latter stage performs operations in such a way that the B component is reduced, for example, in video which appears bluish in white regions.

This adjustment of each color includes changing the coefficients of the sensor or the linear matrix as the latter stage.

2) In-block sum of absolute difference values

Focusing on the G component within the block, the overall value for the differences from the adjacent parts of the image is extracted and accumulation is performed within the block.

On the CPU side, the weighting is applied per block based on the information from each block, and the obtained evaluation values can be used for focus adjustment, etc. from the next frame.

The weighing could in general be center-weighted allocation, weighted allocation according to the subject, and so on. If the obtained evaluation value is compared with the previous frame and if the amount continues to be reduced, the evaluation value is increased to adjust the focus, for example, by adjusting the distance between the lens and the sensor toward the opposite direction.

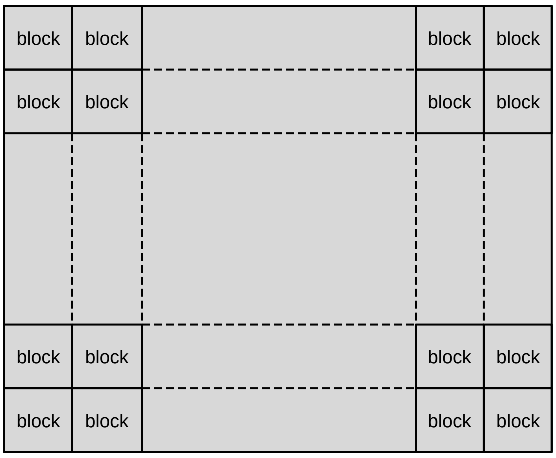


Figure 35.32 Image of Dividing an Image into Blocks Vertically and Horizontally

Figure 35.33 is a block diagram of statistics. The result of AE/AWB statistical calculations for per block is 8 bits × 3 = 3 bytes, respectively, and that of the AF statistical calculations is 1 byte for G only, so a total of 4 bytes of data are output.

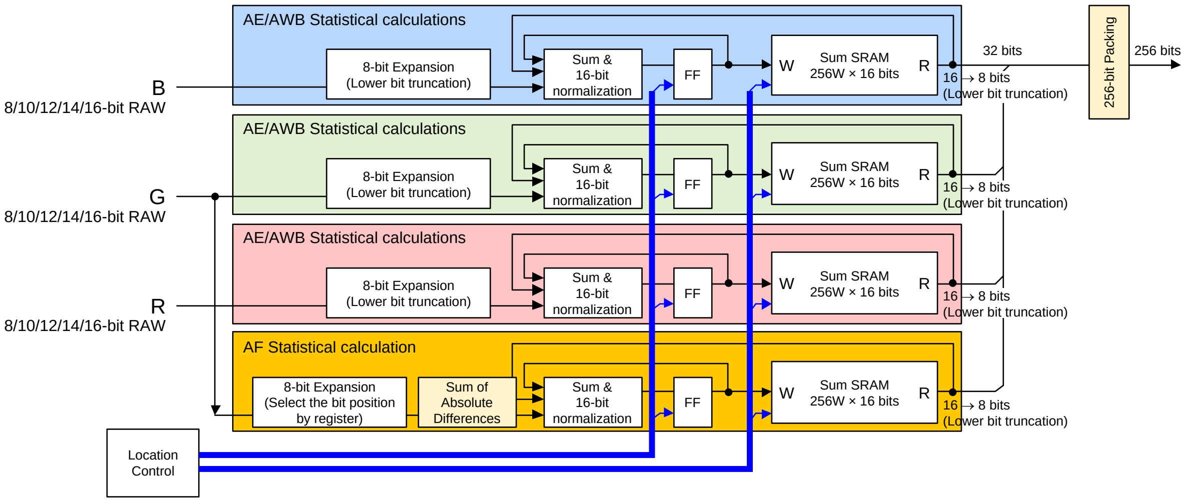


Figure 35.33 Statistics Block Diagram

(1) RGB cumulative addition

The STUNIT[1:0] bit fields in each of the ICnSTIC1 registers are used to select the pixel (block) sizes for statistical calculations from among four values, 16×16, 32×32, 64×64, and 128×128. The target pixels when the setting is 16×16 are described in **(a) Target pixels when the block size is selected as 16×16 (ICnSTIC1.STUNIT[1:0] = 0b)** and the methods of calculation for each block-size setting are described in **(b) Methods of calculation for each block-size setting**.

(a) Target pixels when the block size is selected as 16×16 (ICnSTIC1.STUNIT[1:0] = 0b)

G component: 128 pixels are present in the area of 16×16

R and B components: 32 pixels are present in the area of 16×16

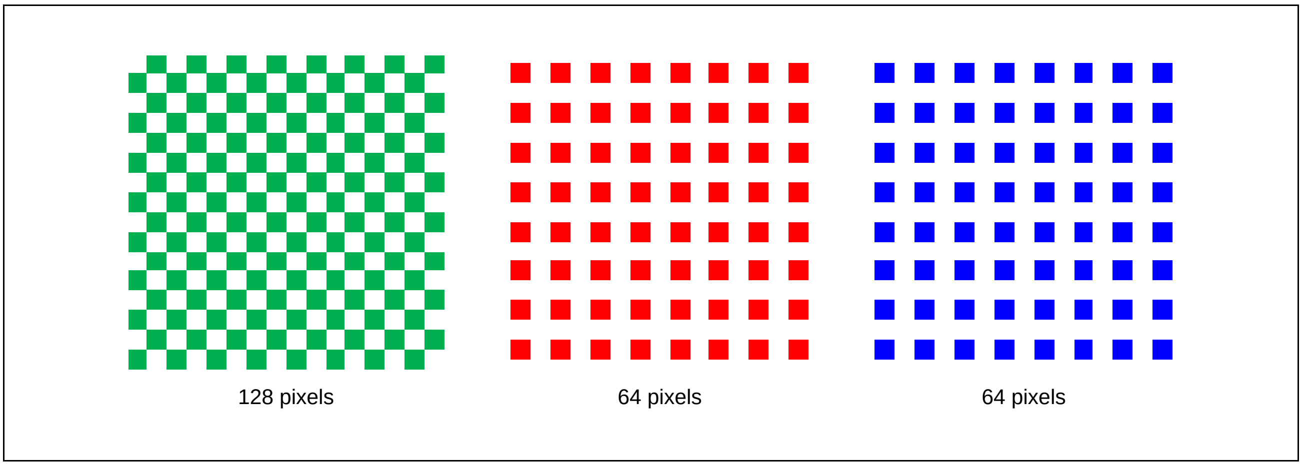


Figure 35.34 Target Pixels when the Block Size is Selected as 16×16

(b) Methods of calculation for each block-size setting

[G component]

- When the block size is 16×16: 8 bits × 128 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 32×32: 8 bits × 512 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 64×64: 8 bits × 2048 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 128×128: 8 bits × 8192 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.

[R and B components]

- When the block size is 16×16: 8 bits × 64 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 32×32: 8 bits × 256 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 64×64: 8 bits × 1024 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.

- When the block size is 128×128: 8 bits × 4096 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.

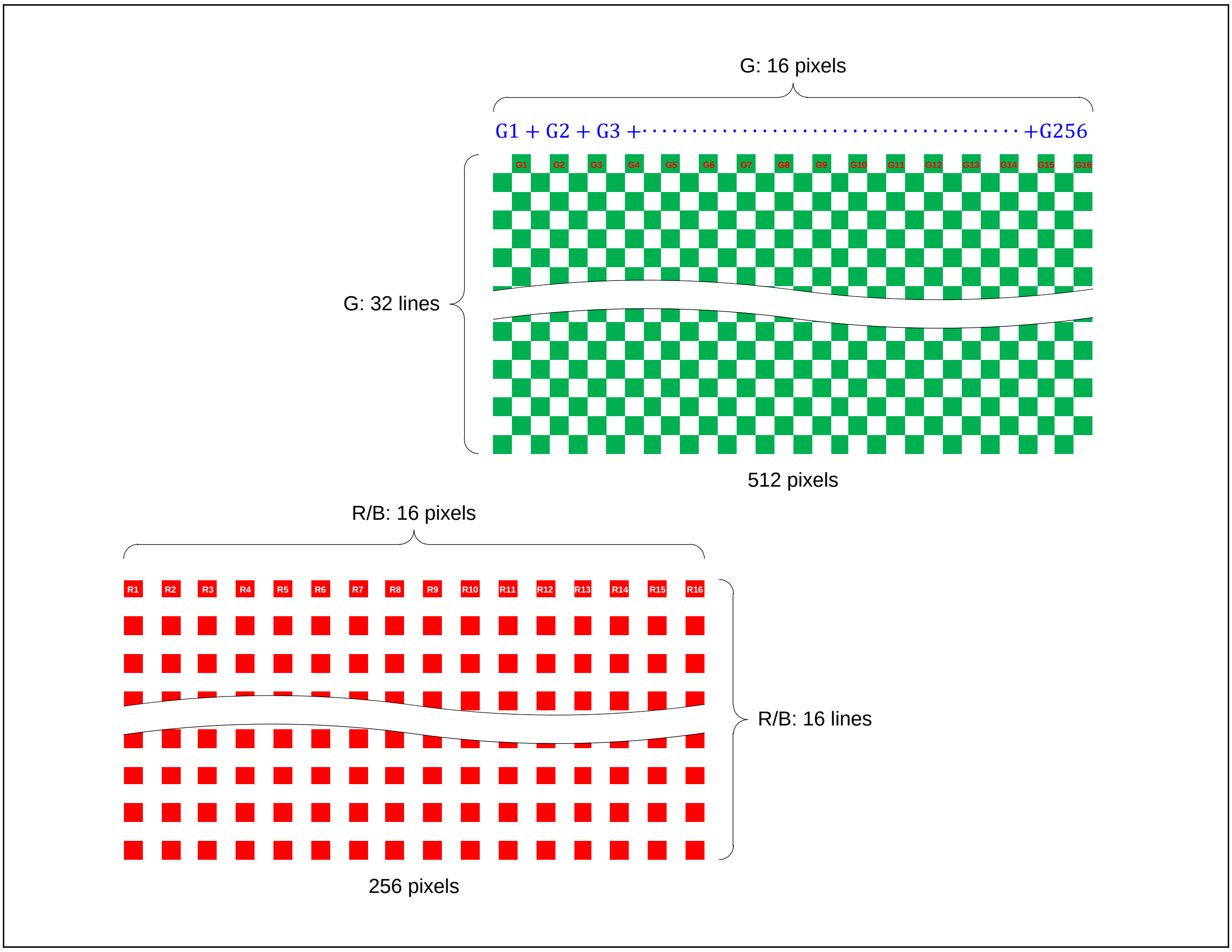


Figure 35.35 Illustration Diagram of Accumulation within the Block when the Block Size is 32×32

(2) Sum of absolute values for adjacent pixels

Set the area for which the sum of absolute values for adjacent pixels is taken with the use of register ICnSTIC1.STUNIT[1:0] to make the calculations in the following ways.

- **In the case of 16×16 pixel unit operations (ICnSTIC1.STUNIT[1:0] = 0b):**
  - For 8-bit data: Up to 16 bits
  - For 10- to 16-bit data: A sum-of-absolute-differences (SAD) operation is performed with the eight higher-order bits (default).
  - Addition can also be performed with the eight lower-order bits by shifting the bits by setting register ICnSTIC1.STSADPOS[3:0].

- **Methods of calculation for the setting area:**

[G component calculation]

- When the block size is 16×16: Adjacent differences × 112 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 32×32: Adjacent differences × 480 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 64×64: Adjacent differences × 1984 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 128×128: Adjacent differences × 8064 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.

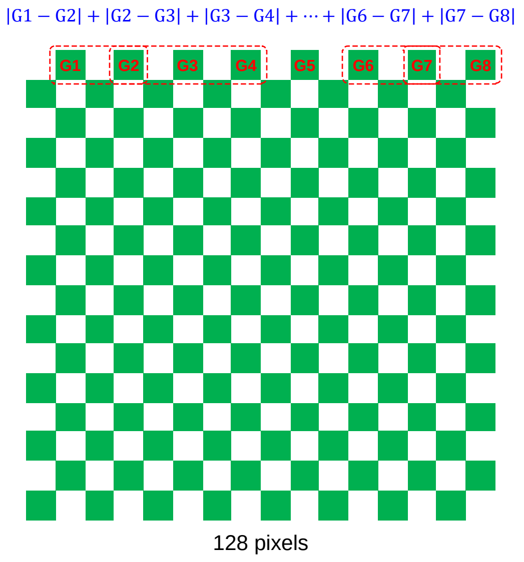


Figure 35.36 Illustration Diagram of Sum of Absolute Values for Adjacent Pixels when the Block Size is 16×16

(3) Statistics data generated image

The following describes the amounts of data in the operations when a 1920×1080 pixel area is set as 16×16 blocks and as 128×128 blocks for the statistics function. The numerals in the figures below represent the block numbers. In processing of the right end or bottom of the image, if the input image size is greater than the size specified as the number of blocks in the horizontal direction, that difference does not fall within the target range for arithmetic processing. This also applies in the vertical direction.

● In the case of operations for a 1920×1080 pixel area as 16×16 blocks:

As shown in **Figure 35.37**, the calculated values for each of the blocks are transferred sequentially. A total of 31 Kbytes of data are transferred for one frame. For the format of data for transmission, see **Table 35.14**.

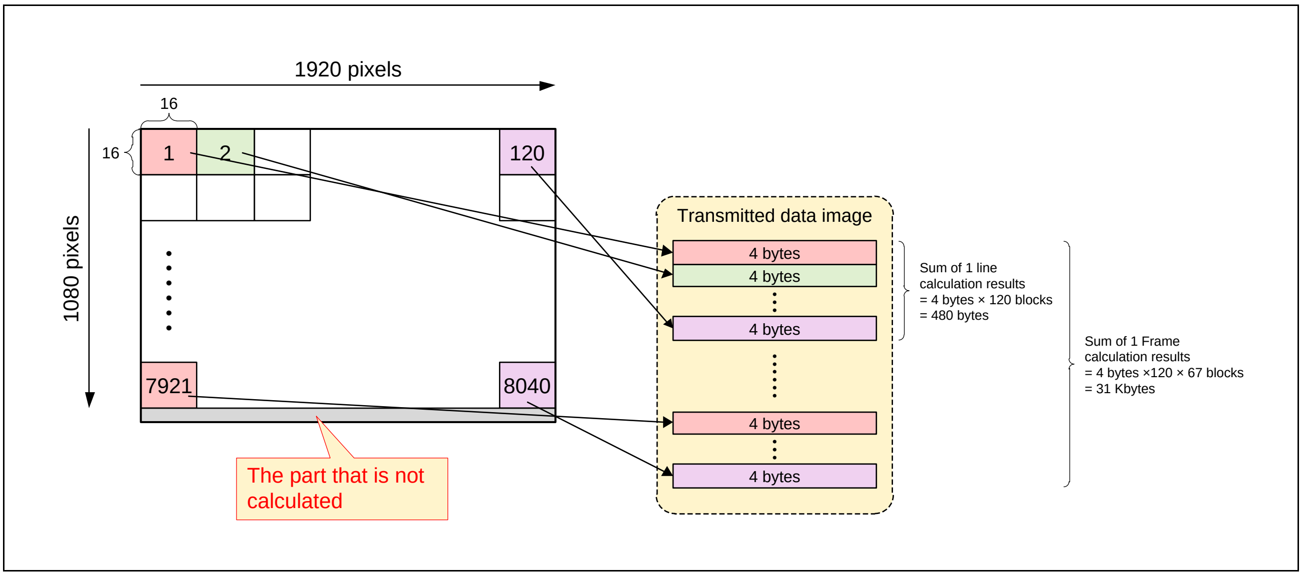


Figure 35.37 Illustration Diagram of the Amounts of Data in the Operations when a 1920×1080 Pixel Area is Set as 16×16 Blocks

- **In the case of operations for a 1920×1080 pixel area as 128×128 blocks:**

The calculated values for the corresponding blocks are sequentially transferred. A total of 480 bytes of data are transferred for one frame.

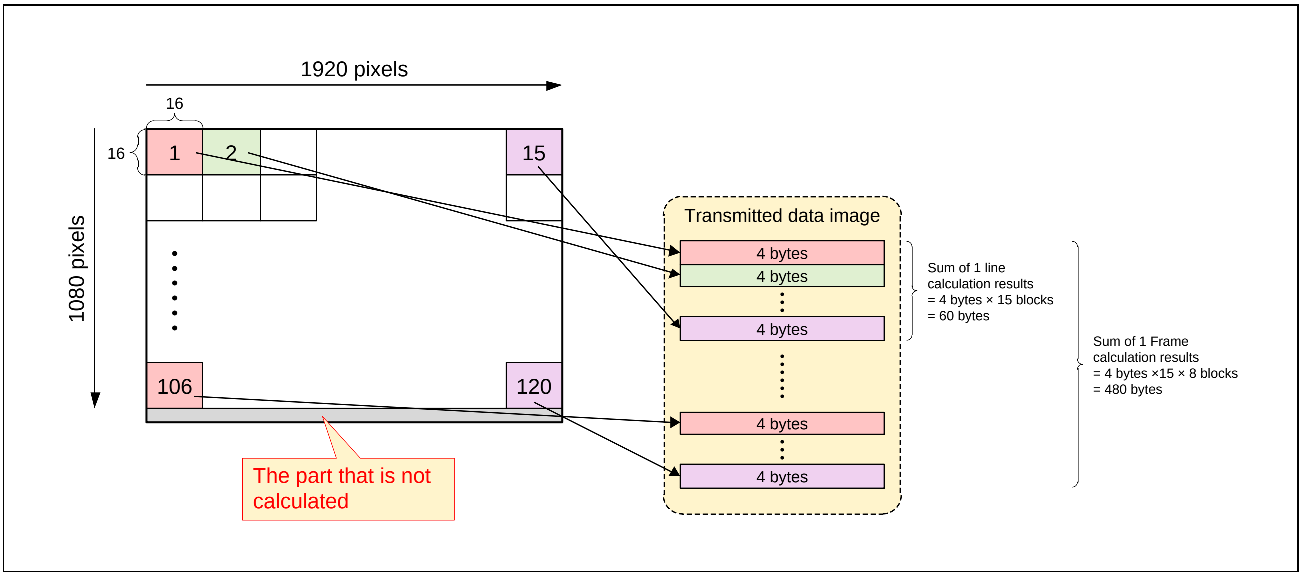


Figure 35.38 Illustration Diagram of the Amounts of Data in the Operations when a 1920×1080 Pixel Area is Set as 16×16 Blocks

(4) Horizontal pixels exceed 2048

Statistics can only detect up to 2048 pixels horizontally. If horizontal pixel exceeds 2048 pixels, the offset setting (ICnSTIC2.STHPOS[8:0]) can be used to detect the center pixel and calculate Statistics.

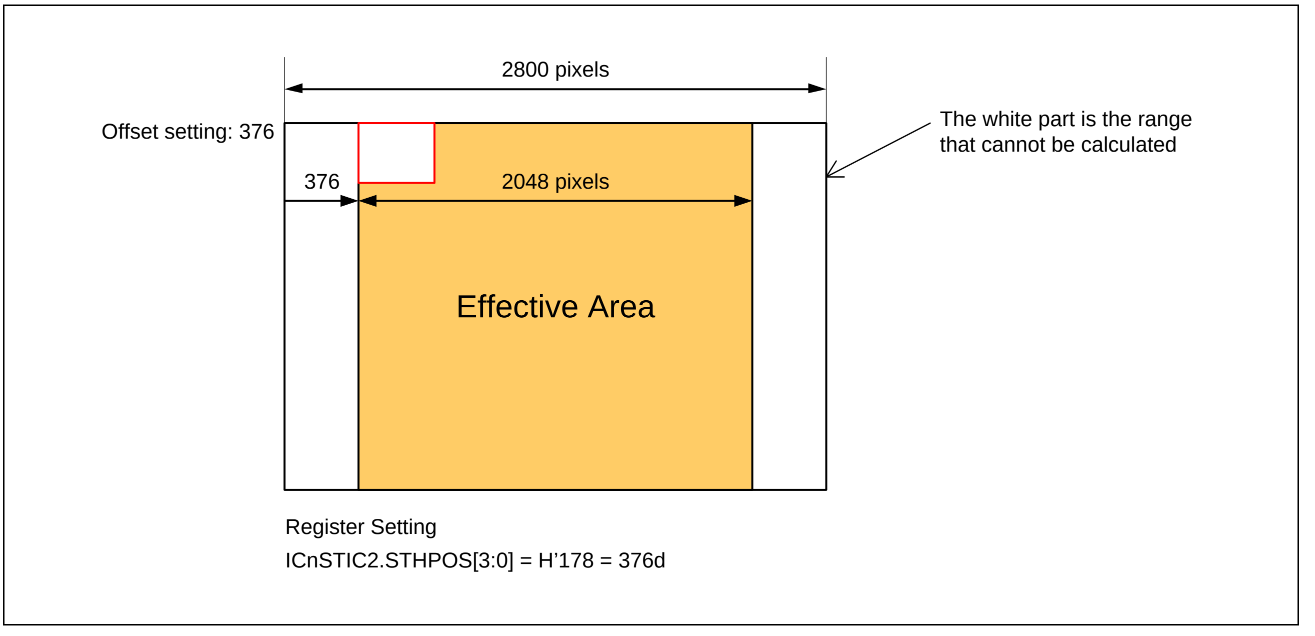


Figure 35.39 Statistics Calculation Example at 2800 Pixels Width

35.2.4.19 Pattern Generator (for Debugging)

A test image (YUV422 only) can be generated for use as an MIPI CSI-2 input by setting the following registers. This function is used for debugging to identify where the problem is originated (in the sensor or in the internal settings). Two modes are implemented for this pattern (monochrome or pattern). See **Section 35.3** for the setting procedure.

TIEN = ICnTICTRL1.TIEN: Enables generation of a test image pattern.
TIMODE = ICnTICTRL1.TIMODE: Selects a test image generation pattern (monochrome or pattern).

TIPTNY1[3:0] = ICnTICTRL1.TIPTNY1[3:0]: Bit settings for the pixel (Y) counter.
TIPTNU1[3:0] = ICnTICTRL1.TIPTNU1[3:0]: Bit settings for the pixel (U) counter.
TIPTNV1[3:0] = ICnTICTRL1.TIPTNV1[3:0]: Bit settings for the pixel (V) counter.

TIRATE[4:0] = ICnTICTRL1.TIRATE[4:0]: Specify a test image generation rate.

TIPTNY2[7:0] = ICnTICTRL2.TIPTNY2[7:0]: Specify V for the entire-screen same-color test image pattern.
TIPTNU2[7:0] = ICnTICTRL2.TIPTNU2[7:0]: Specify U for the entire-screen same-color test image pattern.
TIPTNV2[7:0] = ICnTICTRL2.TIPTNV2[7:0]: Specify Y for the entire-screen same-color test image pattern.

TIPPL[11:0] = ICnTISIZE1.TIPPL[11:0]: Number of valid pixels per line in the test image pattern.
TIN[11:0] = ICnTISIZE2.TIN[11:0]: Valid period (number of lines) per frame in the test image pattern.
TIM[11:0] = ICnTISIZE2.TIM[11:0]: Invalid period (number of lines) per frame in the test image pattern.

(1) Image and Settings (Common to Monochrome and Pattern Generation)

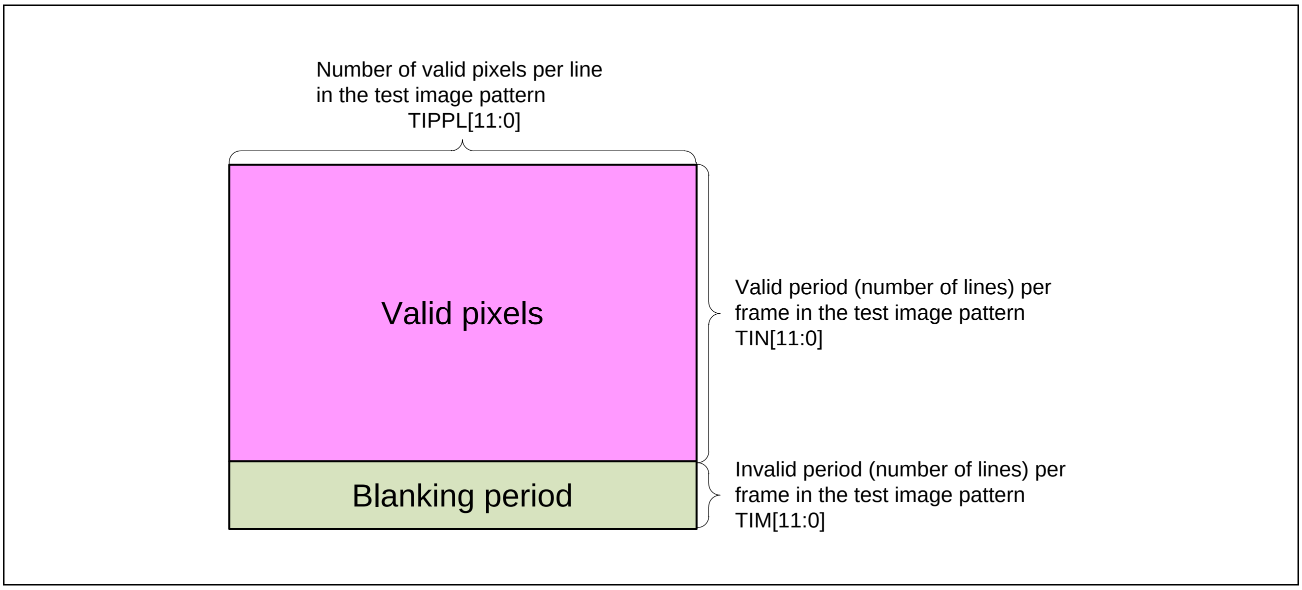


Figure 35.40 Image and Settings

(2) Timing Generation and Settings (Common to Monochrome and Pattern Generation)

(Bits TIRATE[4:0] allow the test image generation rate to be set in the range between 3.783 Gbps and 1.390 Gbps.)

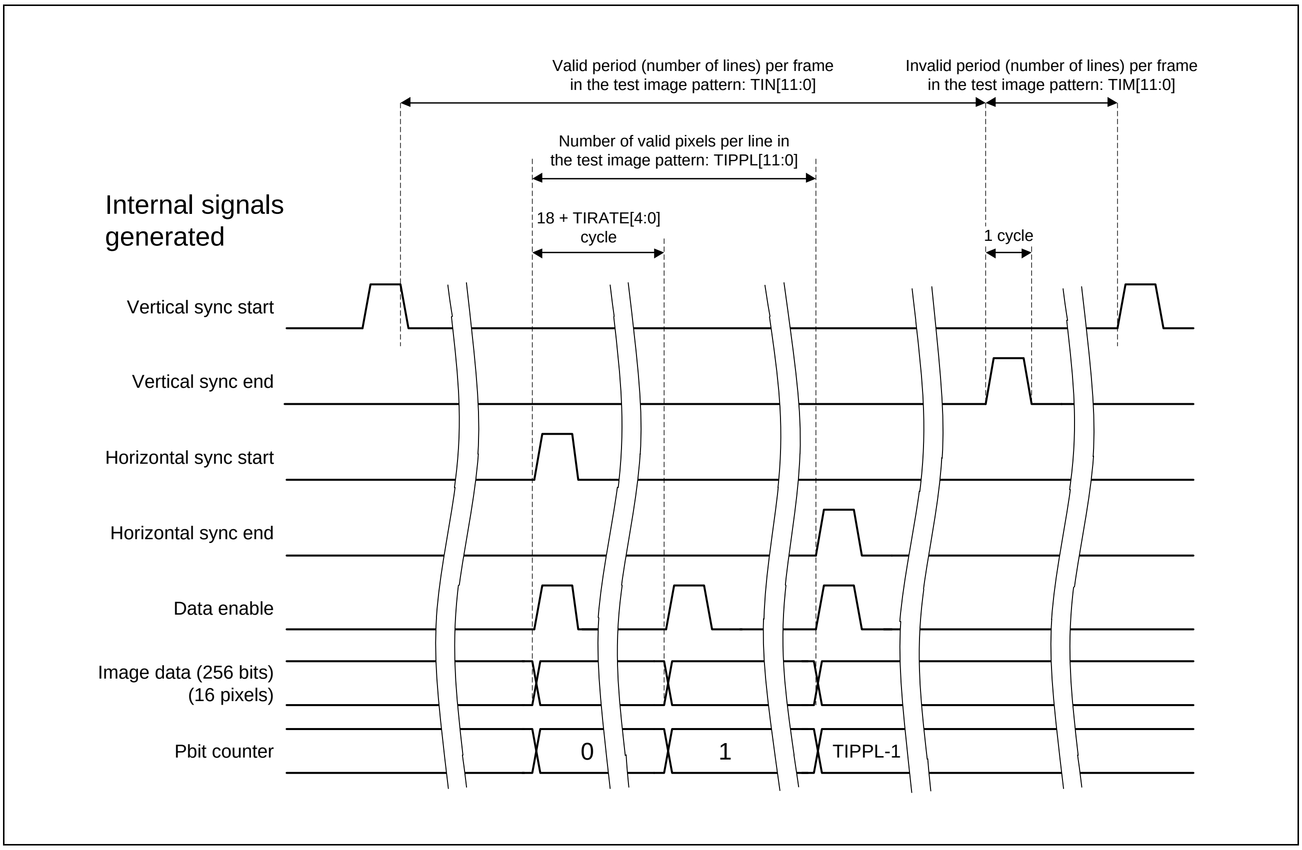
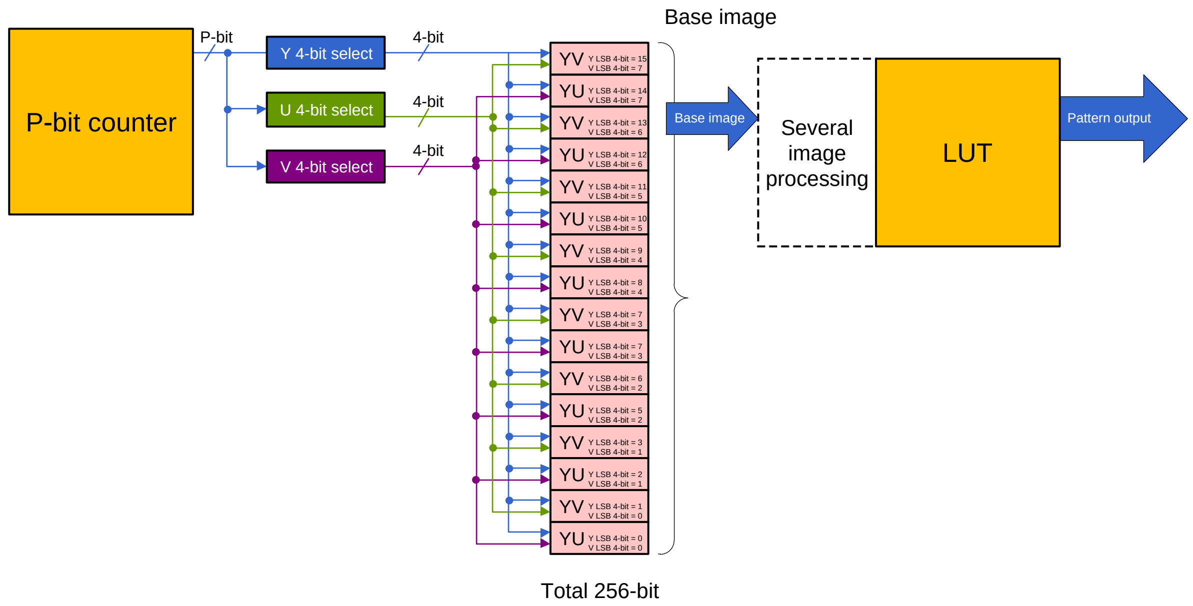


Figure 35.41 Timing Generation and Settings

(3) Data Generation Circuit (in the Case of Pattern Generation)

Image YUV422 8-bit data of 256 bits (16 pixels) is generated as follows (other formats cannot be generated). The 16-bit Pbit counter is incremented in 18 + TIRATE[4:0] (cycles at 266 MHz). The bits of this Pbit counter are selected by the TIPTNV1[3:0], TIPTNU1[3:0], and TIPTNY1[3:0] bits, and then they are packed and sent as follows:

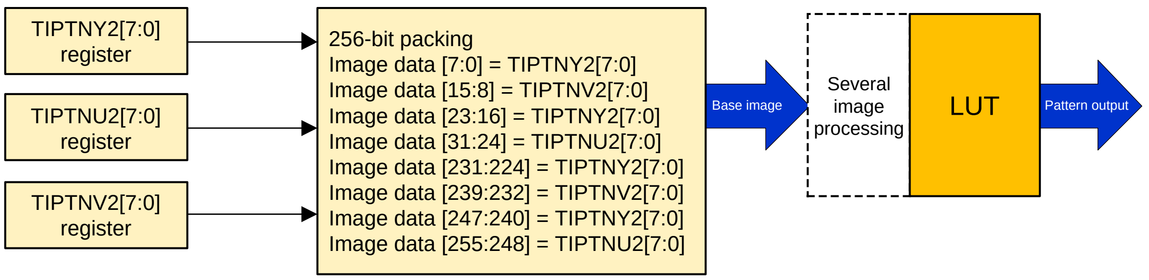


Note: The counter is 16 bits wide.

Figure 35.42 Data Generation Circuit (in the Case of Pattern Generation)

(4) Data Generation Circuit (in the Case of Monochrome Generation)

Image YUV422 8-bit data of 256 bits (16 pixels) is generated in monochrome as follows (other formats cannot be generated). The values set by the TIPTNV2[7:0], TIPTNU2[7:0], and TIPTNY2[7:0] bits are packed and sent as follows:



256-bit packing
Image data [7:0] = TIPTNY2[7:0]
Image data [15:8] = TIPTNV2[7:0]
Image data [23:16] = TIPTNY2[7:0]
Image data [31:24] = TIPTNU2[7:0]
Image data [231:224] = TIPTNY2[7:0]
Image data [239:232] = TIPTNV2[7:0]
Image data [247:240] = TIPTNY2[7:0]
Image data [255:248] = TIPTNU2[7:0]

Figure 35.43 Data Generation Circuit (in the Case of Monochrome Generation)

(5) Generated Image Data Example (in the Case of Pattern Generation)

VGA (640x480): TIEN = 1b, TIMODE = 0b, TIPTNY1[3:0] = 0000b, TIPTNU1[3:0] = 0100b, TIPTNV1[3:0] = 1000b (Offset binary: ICnMC.IBINSEL = 0, ICnDMR.OBINSEL = 0)

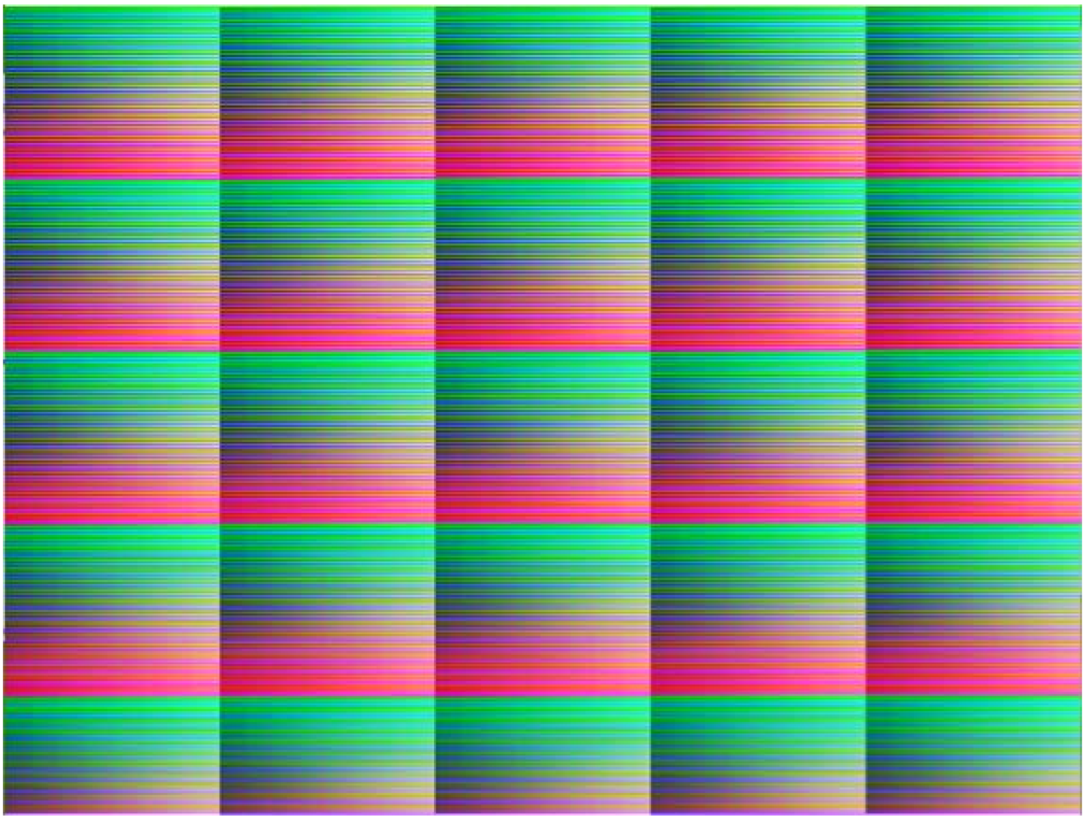


Figure 35.44 Generated Image Data Example (in the Case of Pattern Generation)

(6) Generated Image Data Example (in the Case of Monochrome Generation)

VGA (640x480): TIEN = 1b, TIMODE = 1b, TIPTNY2[7:0] = H’52, TIPTNU2[7:0] = H’5A, TIPTNV2[3:0] = H’F0
(Offset binary: ICnMC.IBINSEL = 0, ICnDMR.OBINSEL = 0)

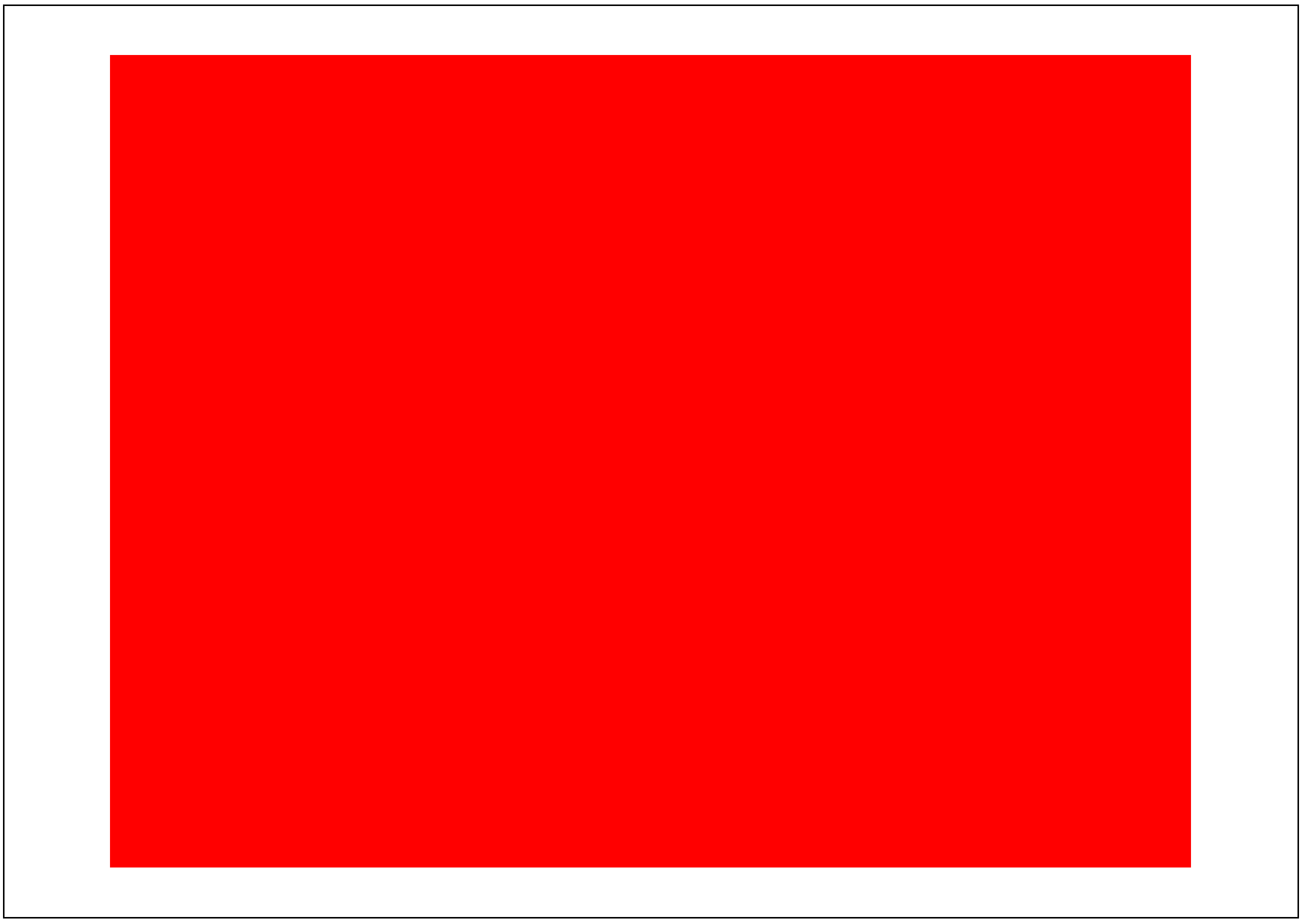


Figure 35.45 Generated Image Data Example (in the Case of Monochrome Generation)

35.2.5 Interrupts

Table 35.18 lists the interrupt sources, registers, and interrupt signals to the interrupt controller.

Table 35.18 Interrupt Sources, Registers, and Interrupt Signals to the Interrupt Controller

Internal Signal Name	Item	Interrupt Source Status/Clear	Interrupt Enable Register Field	Generation Condition
axi_mst_err_int	FIFO overflow (image-related)	CRUnINTS.FOS	CRUnIE.FOE	When the image-related FIFO overflowed
	SLVERR error (image-related)	CRUnINTS.SLVES	CRUnIE.SLVEE	When an SLVERR error occurred in the image-related AXI
	DECERR error (image-related)	CRUnINTS.DECES	CRUnIE.DECEE	When a DECERR error occurred in the image-related AXI
	Frame End Overwrite (image-related)	CRUnINTS.FEOVWS	CRUnIE.FEOVWE	When the Frame End is overwritten in the image-related FIFO
	FIFO-SD overflow (statistics data) (for debugging)	CRUnINTS.SDFOS	CRUnIE1.SDFOE	When Statistics Data FIFO-SD overflows
	SLVERR error (statistics data)	CRUnINTS.SDSLVES	CRUnIE1.SDSLVEE	When the SLVERR error of Statistics Data AXI-SD occurs
	DECERR error (statistics data)	CRUnINTS.SDDECES	CRUnIE1.SDDECEE	When the Statistics Data AXI-SD DECERR error
	Frame End Overwrite (Statistics-related)	CRUnINTS.SDFEOVWS	CRUnIE.SDFEOVWE	When the Frame End is overwritten in the Statistics Data FIFO-SD
image_conv_int	Frame Start	CRUnINTS.SFS	CRUnIE.SFE	When a frame starts *Note*: By using the ICnINTCTRLDECINTE register, whether to generate interrupts in subsampled frames can be specified.
	Frame End	CRUnINTS.EFS	CRUnIE.EFE	When a frame ends *Note*: By using the ICnINTCTRLDECINTE register, whether to generate interrupts in subsampled frames can be specified.
	Scan Line	CRUnINTS.SIS	CRUnIE.SIE	When there is a match with the line number specified in the register (ICnSI.SI[11:0]) *Note*: By using the ICnINTCTRLDECINTE register, whether to generate interrupts in subsampled frames can be specified.
image_conv_err_int	Abnormal word count Value	CRUnINTS.WIS	CRUnIE.WIE	When the word count value set in the register (ICnEWC.EWC[15:0]) did not match the word count value of MIPI
	ITU656 DECERR	CRUnINTS.CES	CRUnIE.CEE	When an ITU656 error was detected

35.2.6 Usage Notes

There are no special usage notes currently.

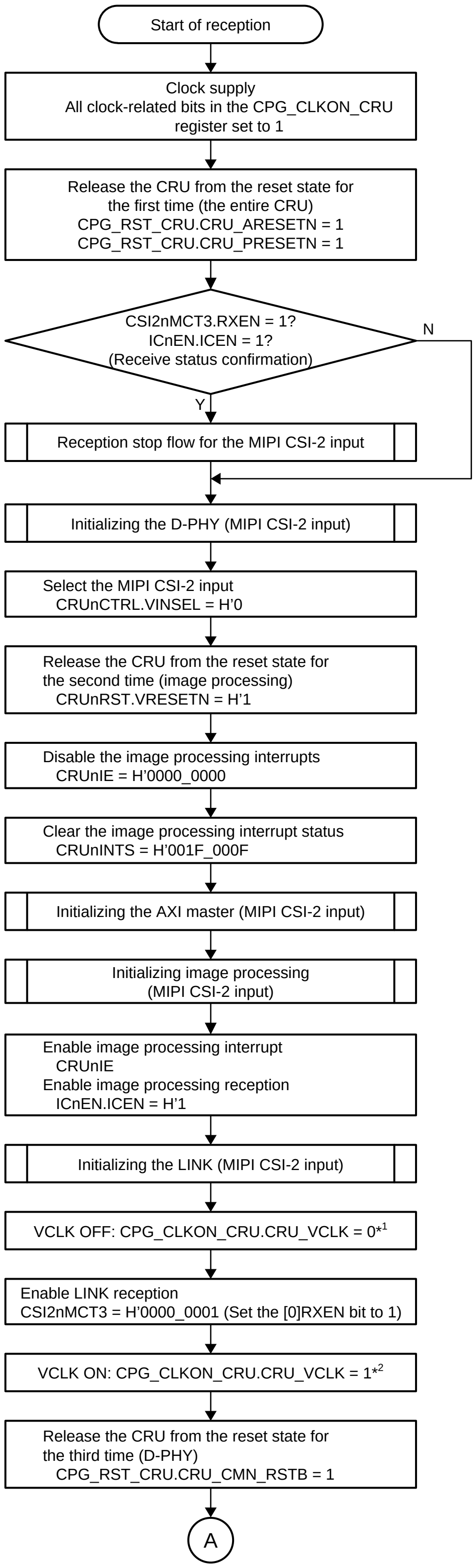
35.3 Operation

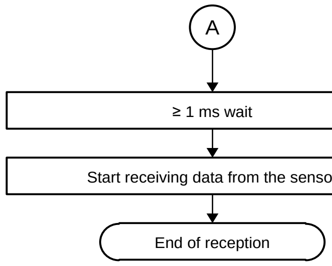
Table 35.19 lists the operations that this section deals with. Follow the procedures described below. If the procedure is not followed, an unexpected behavior may result. The subsections that follow describe each operation.

Table 35.19 List of Operations

Item		Description
Starting reception for the MIPI CSI-2 input		Describes the procedure to start reception for the MIPI CSI-2 input (including the pattern generator).
	Initializing the D-PHY (MIPI CSI-2 input)	Describes the procedure to initialize the D-PHY for the MIPI CSI-2 input.
	Initializing the AXI master (MIPI CSI-2 input)	Describes the procedure to initialize the AXI master for the MIPI CSI-2 input.
	Initializing image processing (MIPI CSI-2 input)	Describes the procedure to initialize image processing for the MIPI CSI-2 input.
	Storing LUT data (both)	Describes the procedure to store LUT data for both the MIPI CSI-2 input and the parallel input.
	Initializing the LINK (MIPI CSI-2 input)	Describes the procedure to initialize the LINK for the MIPI CSI-2 input.
Stopping reception for the MIPI CSI-2 input		Describes the procedure to stop reception for the MIPI CSI-2 input (not including the pattern generator).
	Determining LINK packet stop (MIPI CSI-2 input)	Describes the procedure to determine LINK packet stop for the MIPI CSI-2 input.
	Stopping the D-PHY (MIPI CSI-2 input)	Describes the procedure to stop the D-PHY for the MIPI CSI-2 input.
	Stopping image processing reception (MIPI CSI-2 input)	Describes the procedure to stop image processing reception for the MIPI CSI-2 input.
Changing frame synchronization settings	By setting registers	Describes the procedure to change the frame synchronization settings by setting registers from the CPU for the MIPI CSI-2 input.
	By using the DMAC	Describes the procedure to change the frame synchronization settings by using the DMAC for the MIPI CSI-2 input.
Starting reception for the parallel input		Describes the procedure to start reception for the parallel input (including the pattern generator).
	Initializing the AXI master (parallel input)	Describes the procedure to initialize the AXI master for the parallel input.
	Initializing image processing (parallel input)	Describes the procedure to initialize image processing for the parallel input.
Stopping reception for the parallel input		Describes the procedure to stop reception for the parallel input (not including the pattern generator).
	Determining image processing reception stop (parallel input)	Describes the procedure to determine image processing reception stop for the parallel input.
	Stopping image processing reception (parallel input)	Describes the procedure to stop image processing reception for the parallel input.
Stopping the pattern generator		Describes the stop procedure using the pattern generator.

35.3.1 Starting Reception for the MIPI CSI-2 Input





Note 1. After VCLK OFF setting, confirm whether VCLK is off by reading CPG_CLKMON_CRU.CLK1_ON = 0.

Note 2. After VCLK ON setting, confirm whether VCLK is on by reading CPG_CLKMON_CRU.CLK1_ON = 1.

Figure 35.46 Reception Start Flow for the MIPI CSI-2 Input

35.3.1.1 Initializing the D-PHY (MIPI CSI-2 Input)

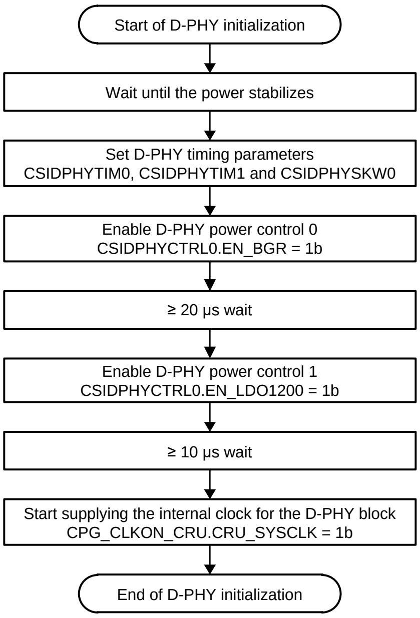


Figure 35.47 Initializing the D-PHY (MIPI CSI-2 Input)

35.3.1.2 Initializing the AXI Master (MIPI CSI-2 Input)

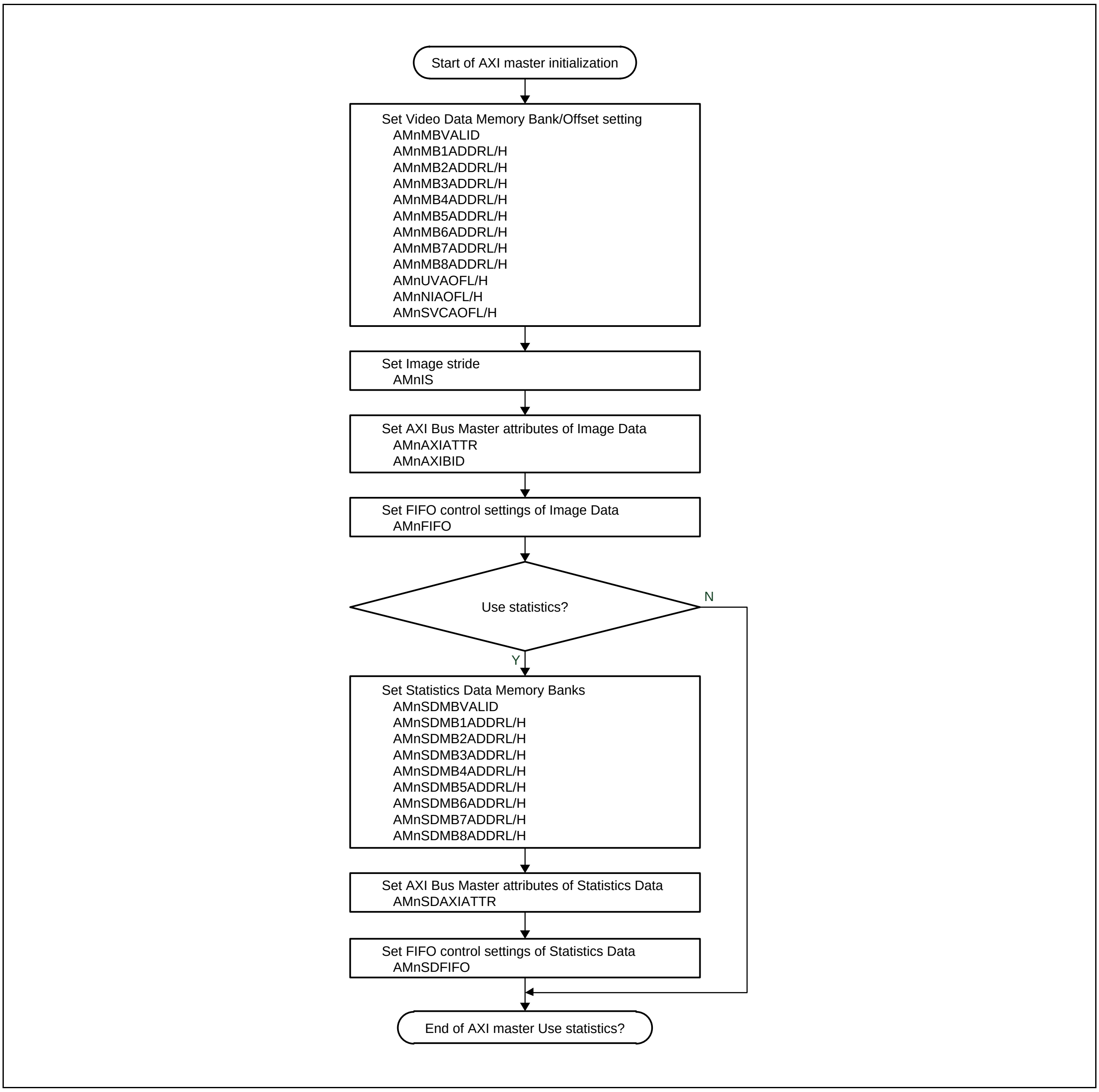
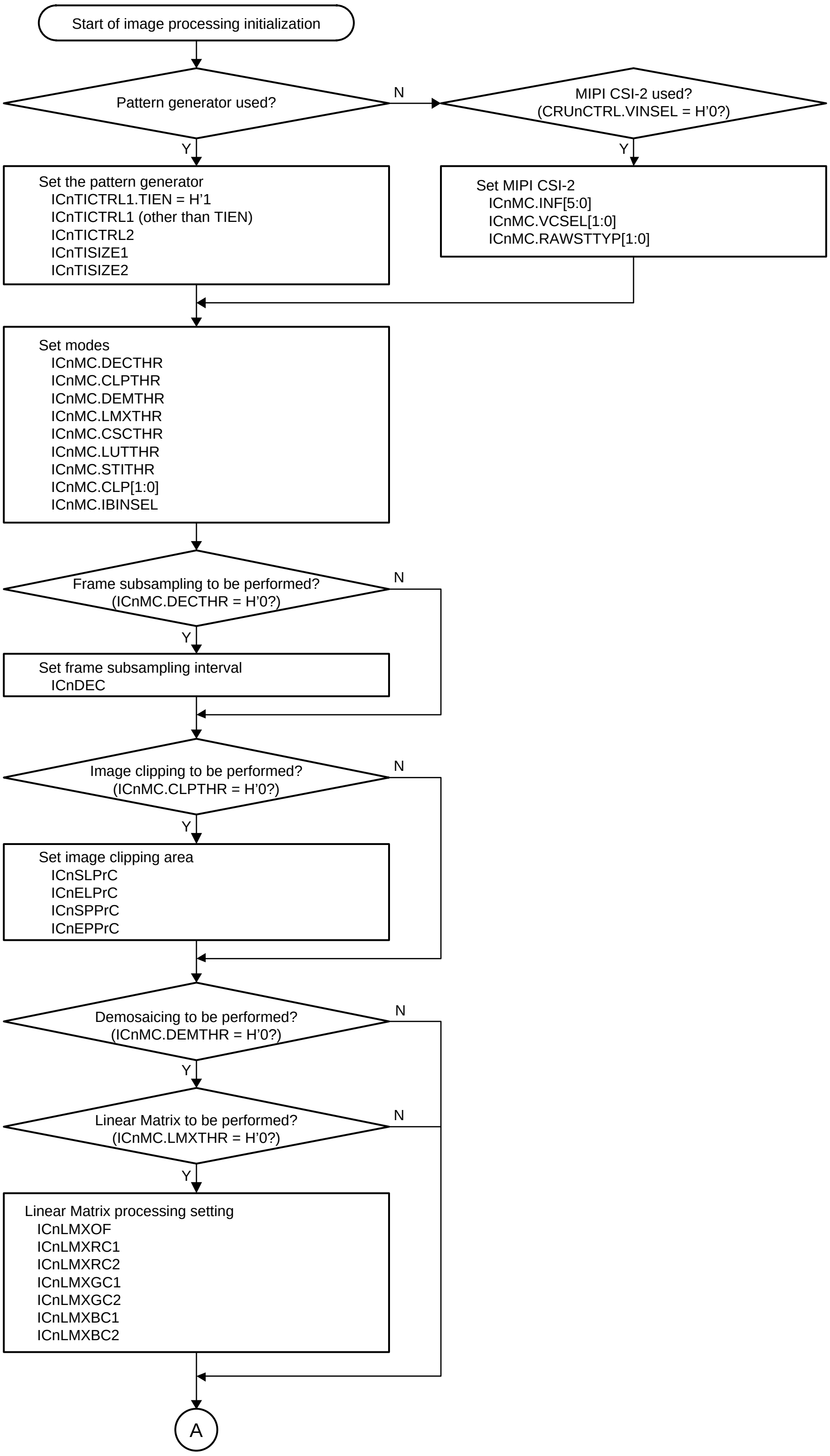


Figure 35.48 Initializing the AXI Master (MIPI CSI-2 Input)

35.3.1.3 Initializing Image Processing (MIPI CSI-2 Input)



Note: For the step processing using generator.
See **Section 35.3.7**

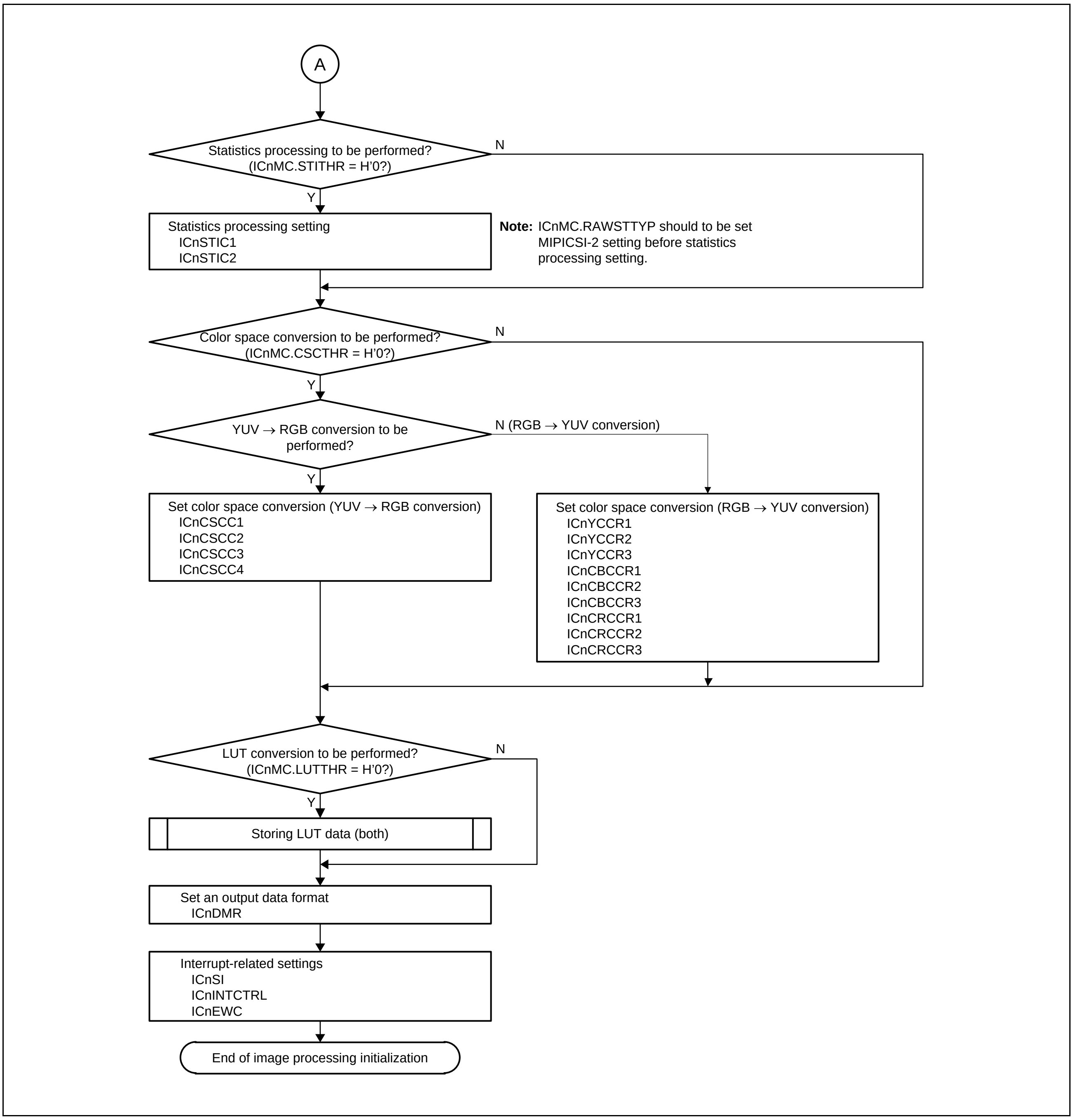


Figure 35.49 Initializing Image Processing (MIPI CSI-2 Input)

35.3.1.4 Storing LUT Data (Both)

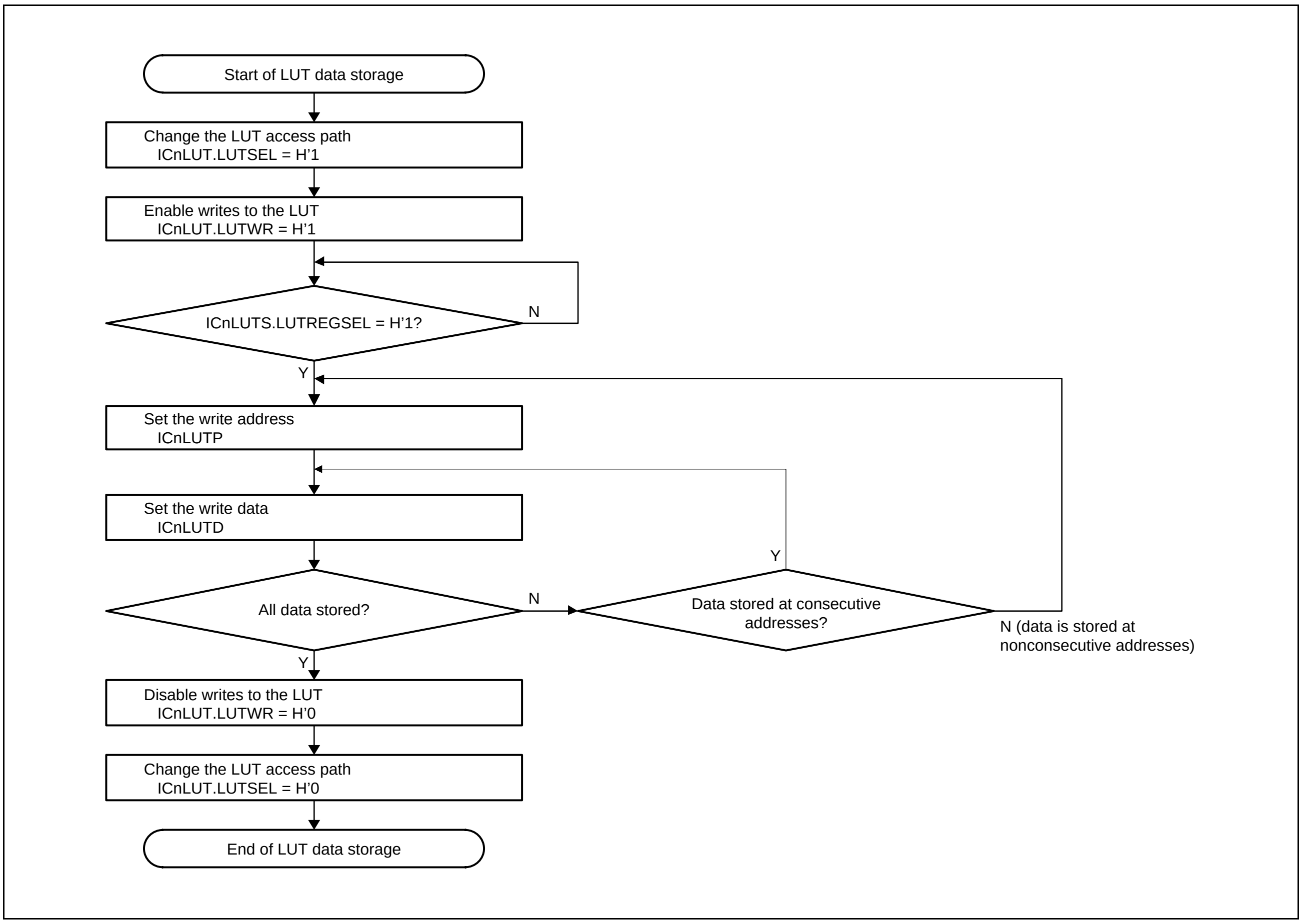
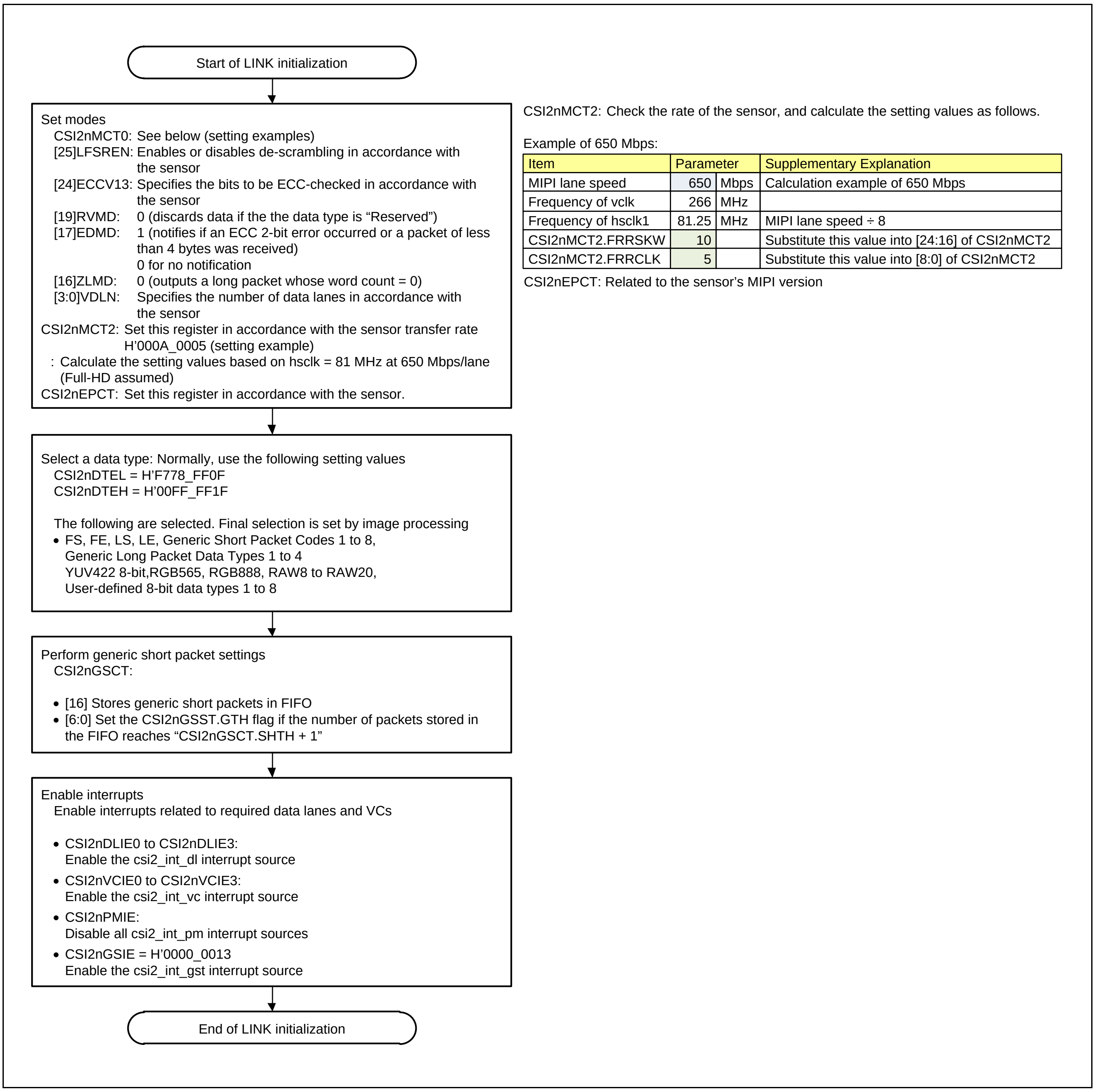


Figure 35.50 Storing LUT Data (Both)

35.3.1.5 Initializing the LINK (MIPI CSI-2 Input)



Set modes
CSI2nMCT0: See below (setting examples)
[25]LFSREN: Enables or disables de-scrambling in accordance with the sensor
[24]ECCV13: Specifies the bits to be ECC-checked in accordance with the sensor
[19]RVMD: 0 (discards data if the the data type is “Reserved”)
[17]EDMD: 1 (notifies if an ECC 2-bit error occurred or a packet of less than 4 bytes was received)
0 for no notification
[16]ZLMD: 0 (outputs a long packet whose word count = 0)
[3:0]VDLN: Specifies the number of data lanes in accordance with the sensor
CSI2nMCT2: Set this register in accordance with the sensor transfer rate H’000A_0005 (setting example)
: Calculate the setting values based on hsclk = 81 MHz at 650 Mbps/lane (Full-HD assumed)
CSI2nEPCT: Set this register in accordance with the sensor.

CSI2nMCT2: Check the rate of the sensor, and calculate the setting values as follows.

Example of 650 Mbps:

Item	Parameter		Supplementary Explanation
MIPI lane speed	650	Mbps	Calculation example of 650 Mbps
Frequency of vclk	266	MHz	
Frequency of hsclk1	81.25	MHz	MIPI lane speed ÷ 8
CSI2nMCT2.FRRSKW	10		Substitute this value into [24:16] of CSI2nMCT2
CSI2nMCT2.FRRCLK	5		Substitute this value into [8:0] of CSI2nMCT2

CSI2nEPCT: Related to the sensor’s MIPI version

Select a data type: Normally, use the following setting values
CSI2nDTEL = H’F778_FF0F
CSI2nDTEH = H’00FF_FF1F

The following are selected. Final selection is set by image processing
- FS, FE, LS, LE, Generic Short Packet Codes 1 to 8, Generic Long Packet Data Types 1 to 4 YUV422 8-bit,RGB565, RGB888, RAW8 to RAW20, User-defined 8-bit data types 1 to 8

Perform generic short packet settings
CSI2nGSCT:

- [16] Stores generic short packets in FIFO
- [6:0] Set the CSI2nGSST.GTH flag if the number of packets stored in the FIFO reaches “CSI2nGSCT.SHTH + 1”

Enable interrupts
Enable interrupts related to required data lanes and VCs

- CSI2nDLIE0 to CSI2nDLIE3: Enable the csi2_int_dl interrupt source
- CSI2nVCIE0 to CSI2nVCIE3: Enable the csi2_int_vc interrupt source
- CSI2nPMIE: Disable all csi2_int_pm interrupt sources
- CSI2nGSIE = H’0000_0013 Enable the csi2_int_gst interrupt source

Figure 35.51 Initializing the LINK (MIPI CSI-2 Input)

35.3.2 Stopping Reception for the MIPI CSI-2 Input

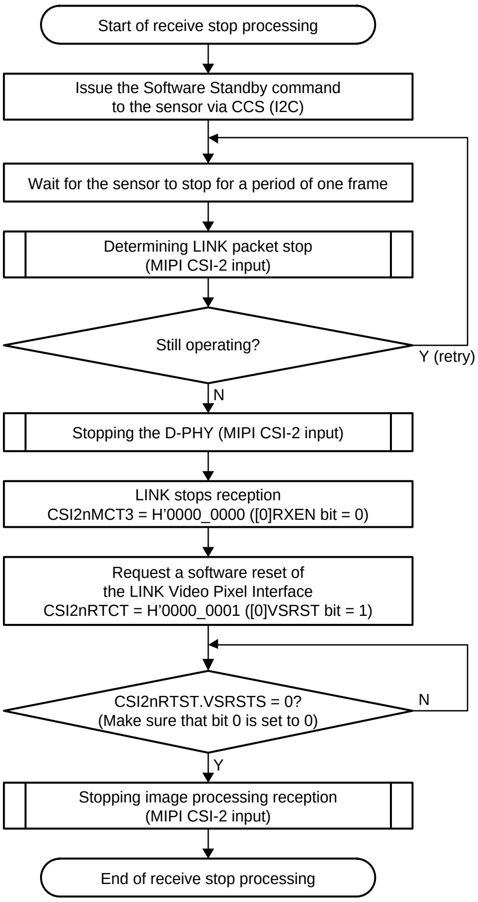


Figure 35.52 Reception Stop Flow for the MIPI CSI-2 Input

35.3.2.1 Determining LINK Packet Stop (MIPI CSI-2 Input)

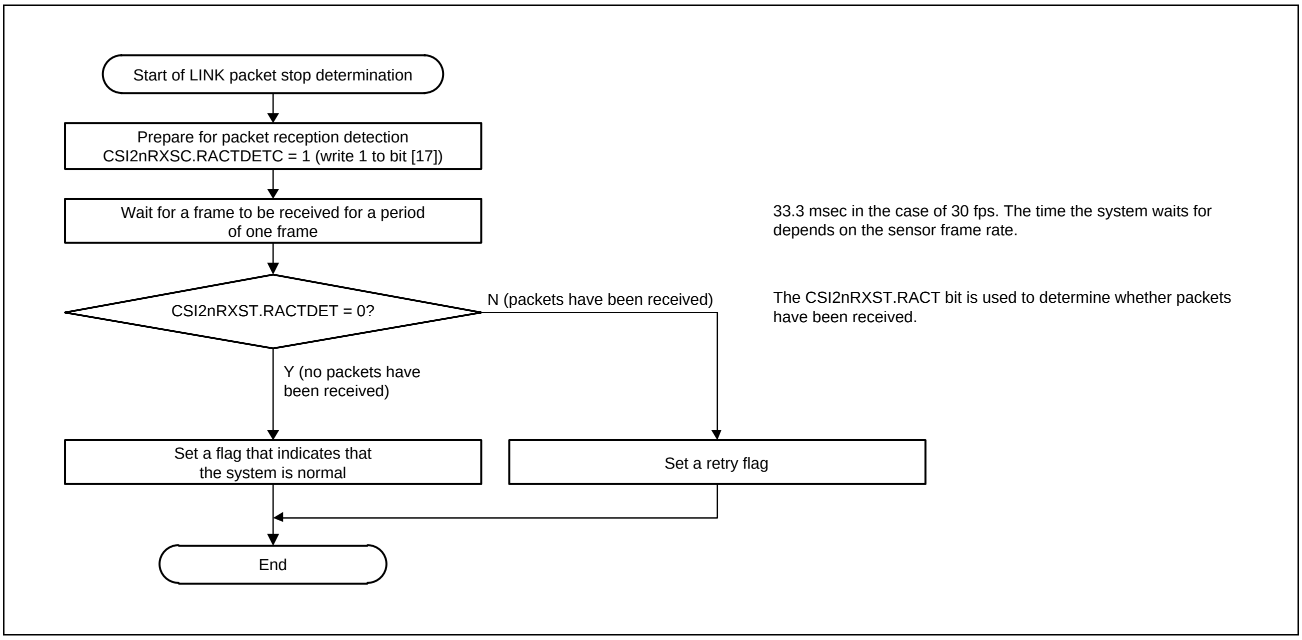
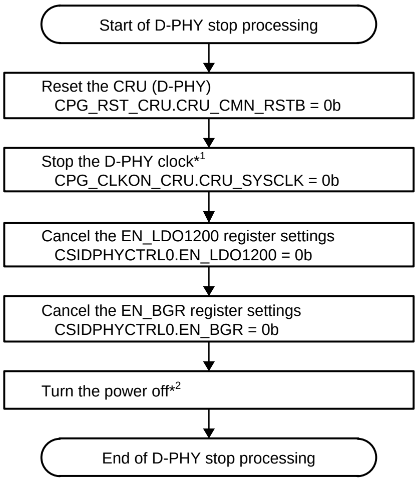


Figure 35.53 Determining LINK Packet Stop (MIPI CSI-2 Input)

35.3.2.2 Stopping the D-PHY (MIPI CSI-2 Input)

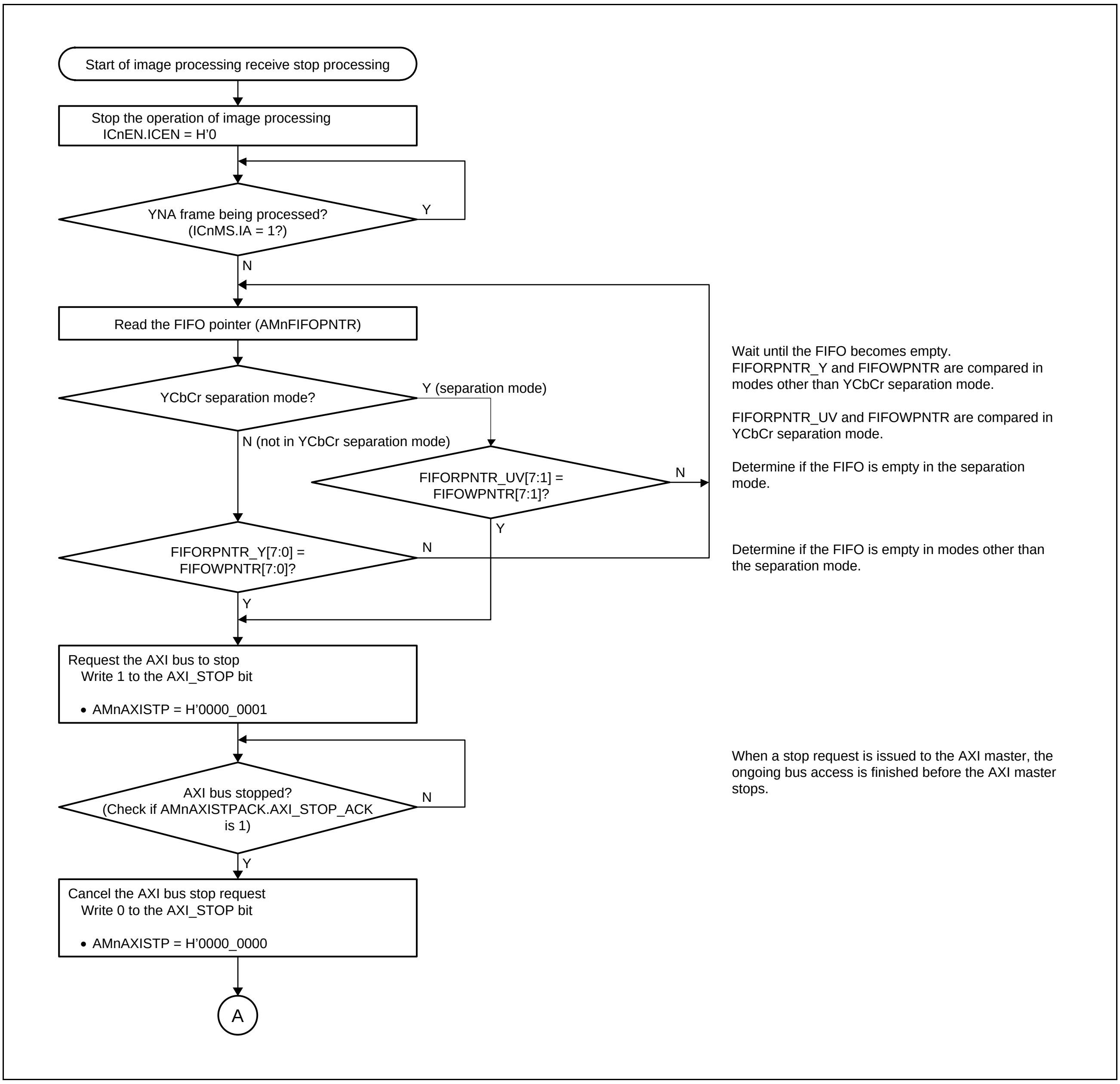


Note 1. This step can be skipped if D-PHY is to be restarted immediately.

Note 2. This step can be skipped if D-PHY is to be restarted immediately.

Figure 35.54 Stopping the D-PHY (MIPI CSI-2 Input)

35.3.2.3 Stopping Image Processing Reception (MIPI CSI-2 Input)



Wait until the FIFO becomes empty.
FIFORPNTR_Y and FIFOWPNTR are compared in modes other than YCbCr separation mode.

FIFORPNTR_UV and FIFOWPNTR are compared in YCbCr separation mode.

Determine if the FIFO is empty in the separation mode.

Determine if the FIFO is empty in modes other than the separation mode.

When a stop request is issued to the AXI master, the ongoing bus access is finished before the AXI master stops.

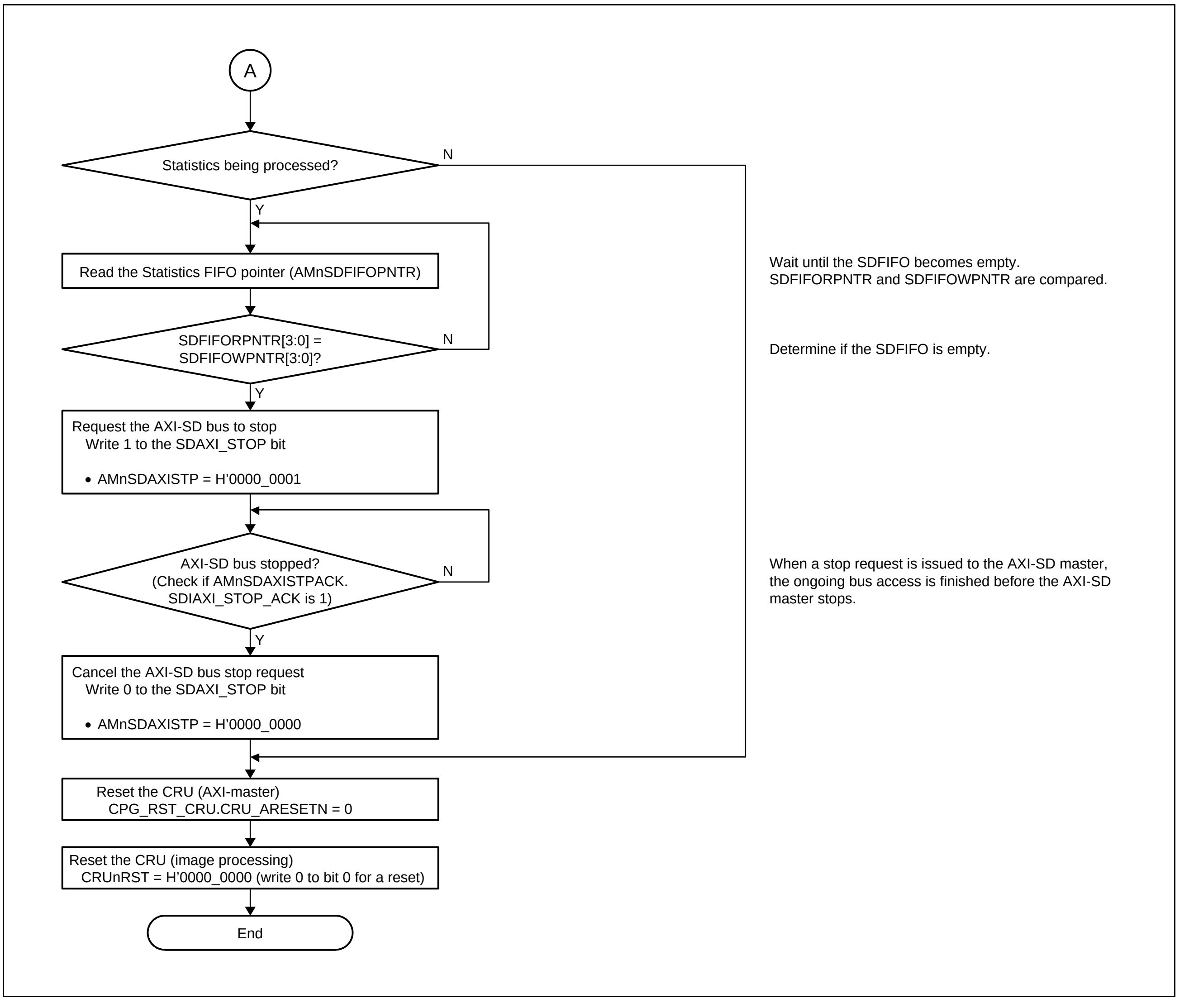


Figure 35.55 Stopping Image Processing Reception (MIPI CSI-2 Input)

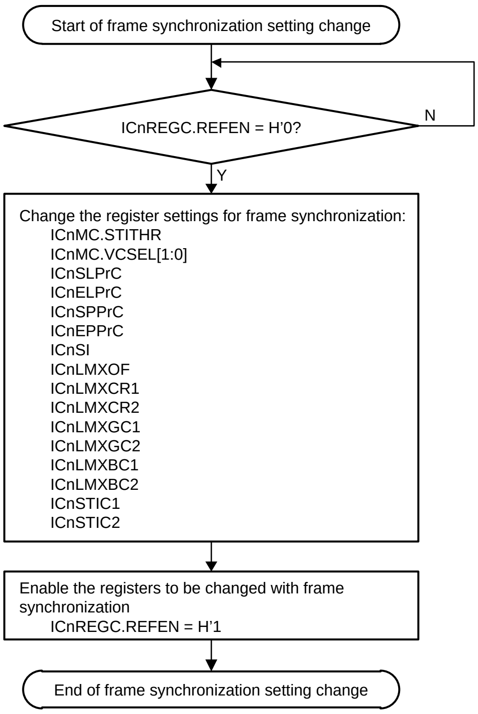
35.3.3 Frame Synchronization Settings

During the CRU is receiving data via the MIPI CSI-2 input, the following registers can only be changed with frame synchronization:

- ICnMC.STITHR
- ICnMC.VCSEL[1:0]
- ICnSLPrC, ICnSPPrC, ICnELPrC, ICnEPPrC
- ICnSI, ICnLMXOF
- ICnLMXRC1, ICnLMXRC2
- ICnLMXGC1, ICnLMXGC2
- ICnLMXBC1, ICnLMXBC2
- ICnSTIC1, ICnSTIC2

This section shows the flows for changing the frame synchronization settings both by setting registers from the CPU and by using the DMAC.

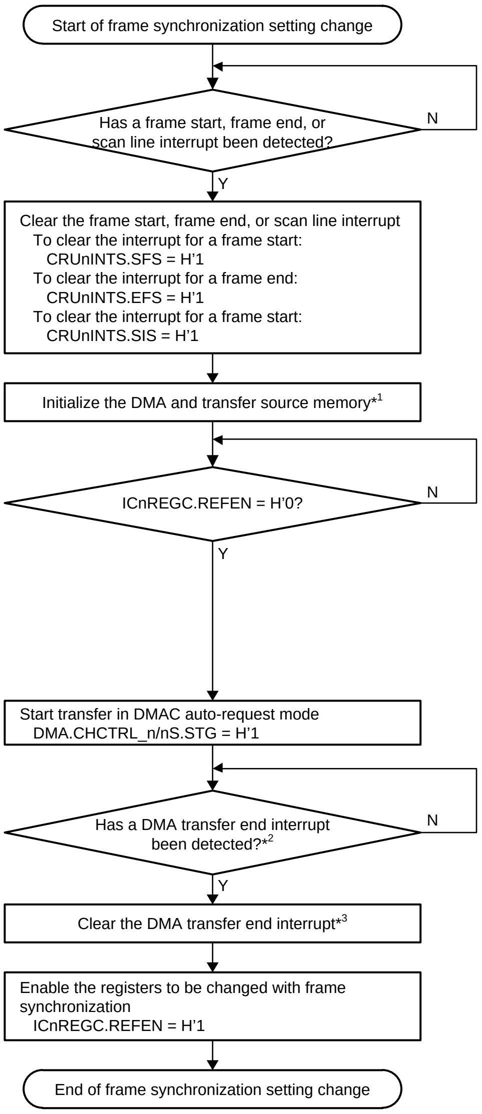
35.3.3.1 Changing Frame Synchronization Settings (by Setting Registers)



Note: This flow for changing the frame synchronization settings is only valid for the MIPI CSI-2 input. It cannot be used for the parallel input.

Figure 35.56 Flow for Changing the Frame Synchronization Settings (by Setting Registers)

35.3.3.2 Changing Frame Synchronization Settings (by Using the DMAC)



NOTES:

1. This flow for changing the frame synchronization settings is only valid for the MIPI CSI-2 input. It cannot be used for the parallel input.
2. In the initial settings when the CRU starts, interrupts for a frame start, frame end and scan line must be set.

   To set an interrupt for a frame start:
   CRUnIE.SFE = H'1

   To set an interrupt for a frame end:
   CRUnIE.EFE = H'1

   To set an interrupt for a frame start:
   CRUnIE.SIE = H'1

   In addition, CRU and DMA interrupts must be enabled by seeing the section about the interrupt controller.

Note 1. Perform initialization by seeing the section about DMA.

- Note that the following register settings are required:
  DMA.CHCFG_n/nS.DAD = H'0: Increment
  DMA.CHCFG_n/nS.SAD = H'0: Increment
  DMA.CHCFG_n/nS.DDS = H'3: 32 bits
- Set the DMA transfer size to 32 bits for both transfer destination and transfer source
- In transfer source memory initialization, store the values that set the following register: (Start address: Up to 64 bytes in ICnMC)
  ICnMC.STITHR, ICnMC.VCSEL[1:0],
  ICnSLPrC, ICnSPPrC, ICnELPrC, ICnEPPrC,
  ICnSI, ICnLMXOF,
  ICnLMXRC1, ICnLMXRC2, ICnLMXGC1, ICnLMXGC2,
  ICnLMXBC1, ICnLMXBC2,
  ICnSTIC1

Note 2. A setting is required in advance (see the sections about DMA and interrupt controller).

Note 3. Clear the interrupt by seeing the section about DMA and interrupt controller.

Figure 35.57 Flow for Changing the Frame Synchronization Settings (by Using the DMAC)

35.3.4 Starting Reception for the Parallel Input

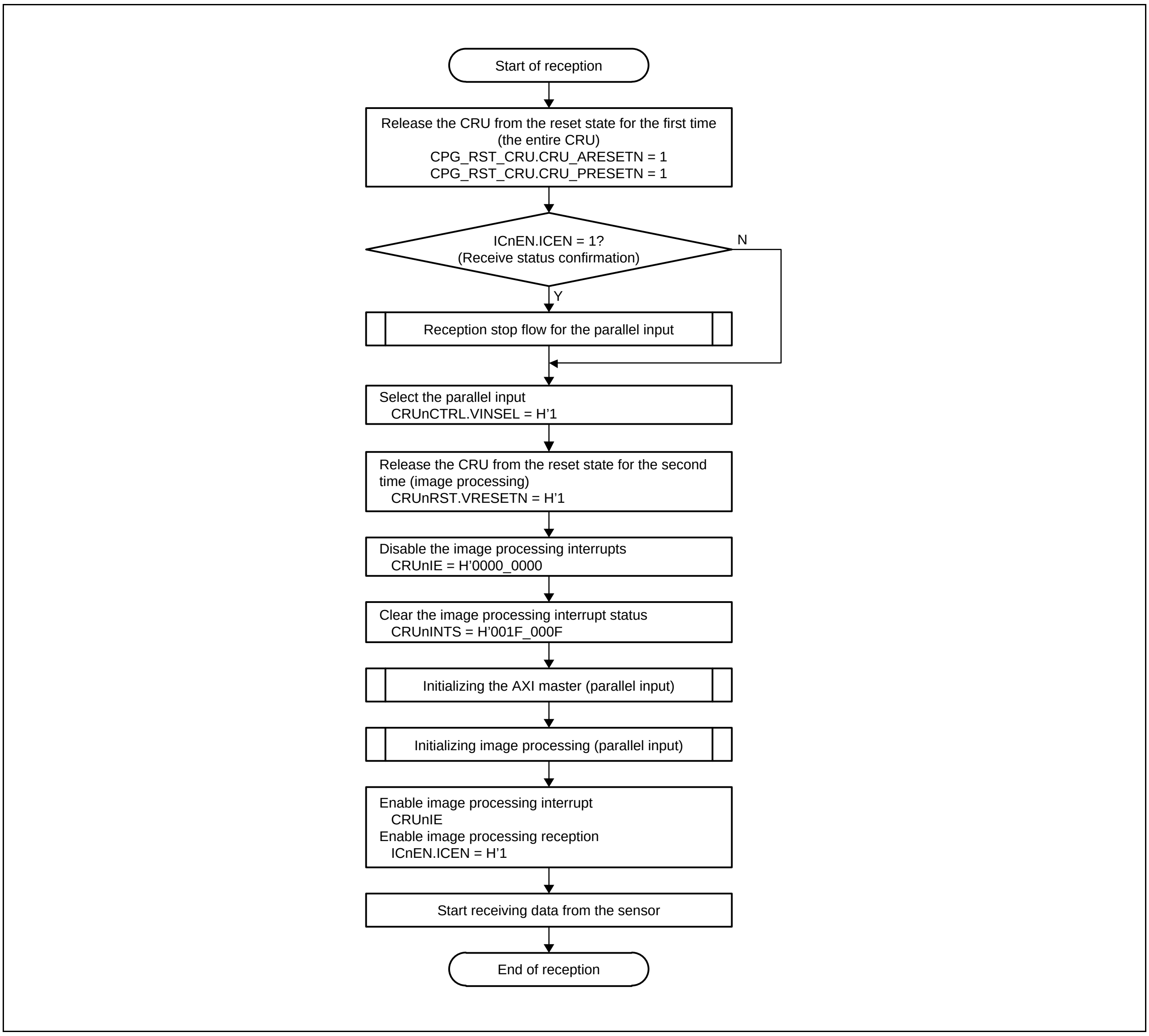


Figure 35.58 Reception Start Flow for the Parallel Input

35.3.4.1 Initializing the AXI Master (Parallel Input)

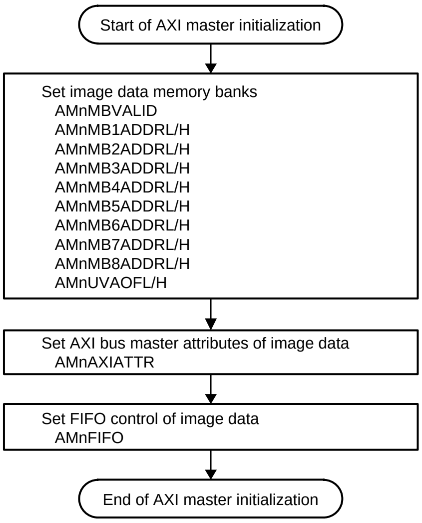
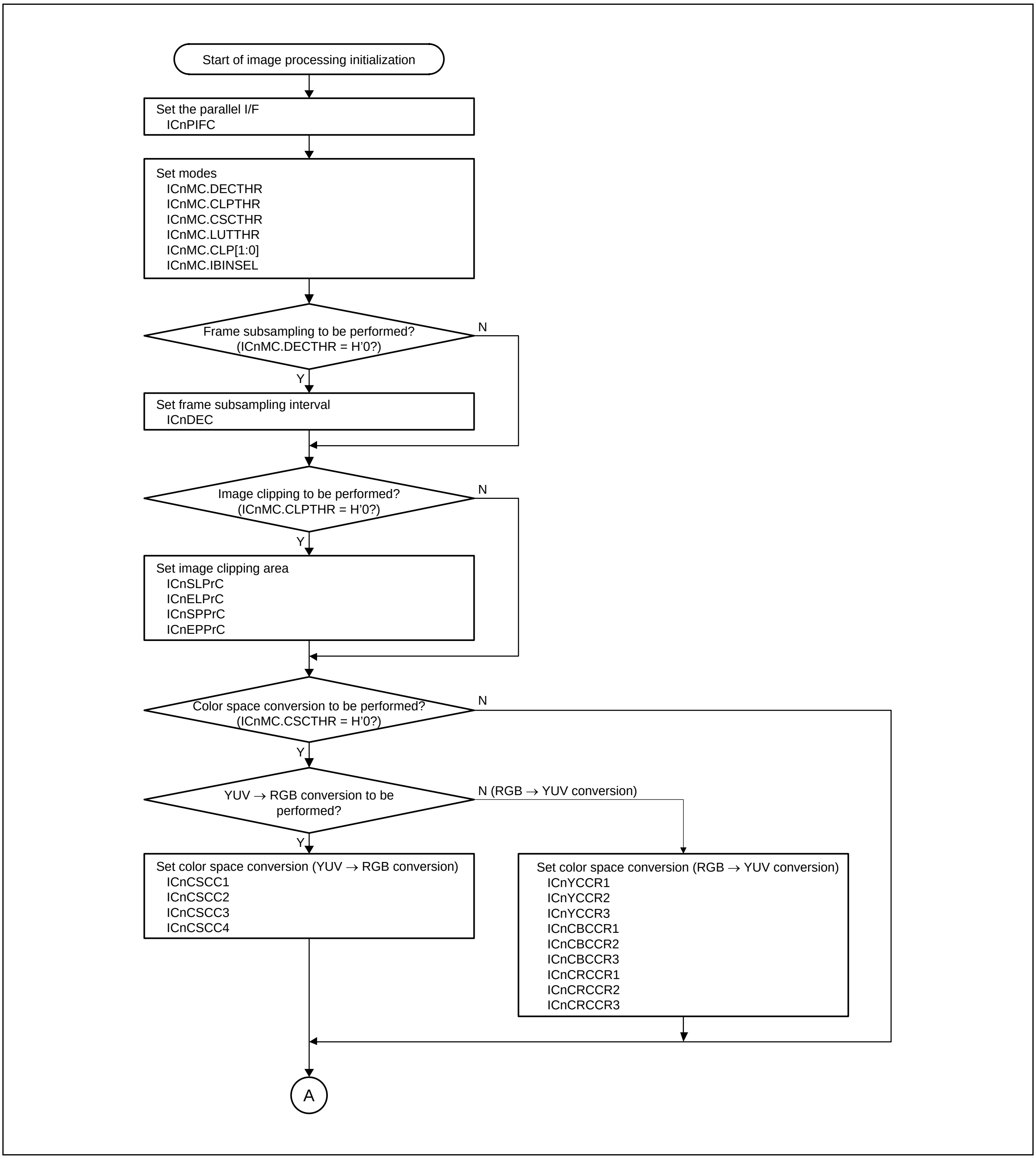
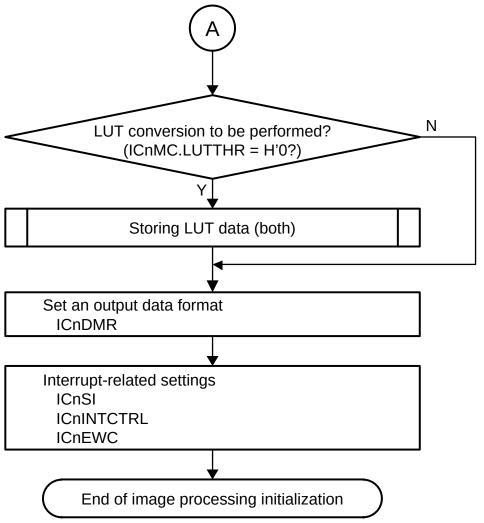


Figure 35.59 Initializing the AXI Master (Parallel Input)

35.3.4.2 Initializing Image Processing (Parallel Input)





Note: When the same value previously specified is used, this flow can be skipped.

Figure 35.60 Initializing Image Processing (Parallel Input)

35.3.4.3 Storing LUT Data (Both)

See **Section 35.3.1.4, Storing LUT Data (Both)**.

35.3.5 Stopping Reception for the Parallel Input

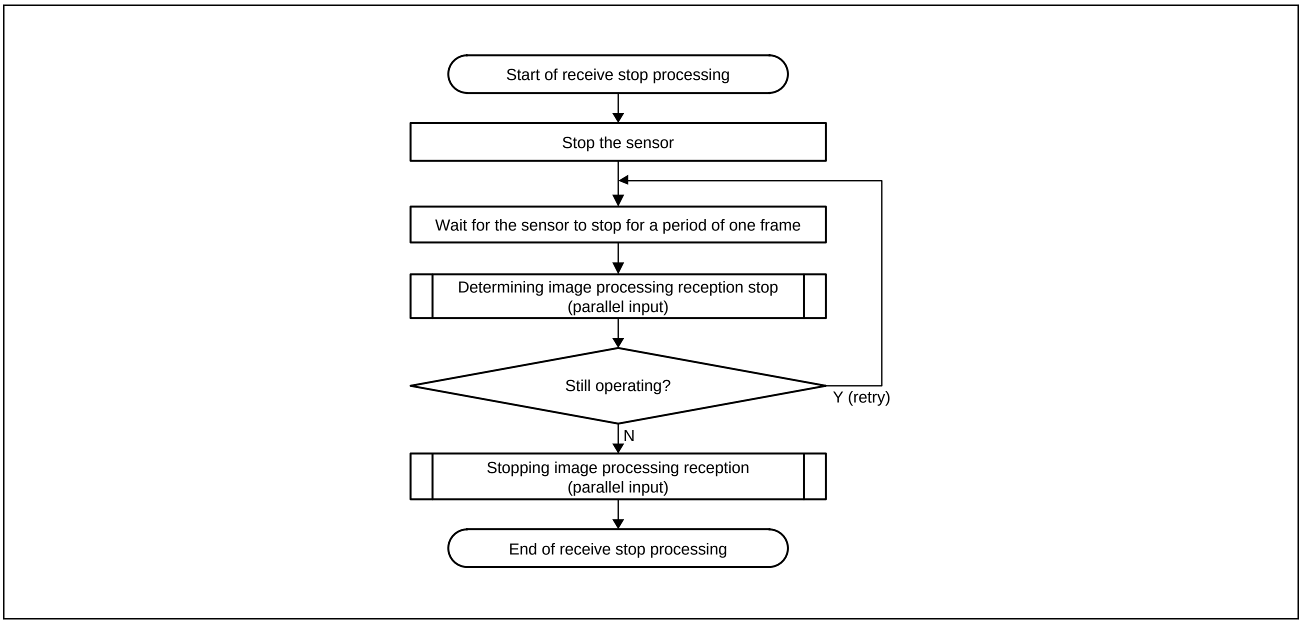


Figure 35.61 Reception Stop Flow for the Parallel Input

35.3.5.1 Determining Image Processing Reception Stop (Parallel Input)

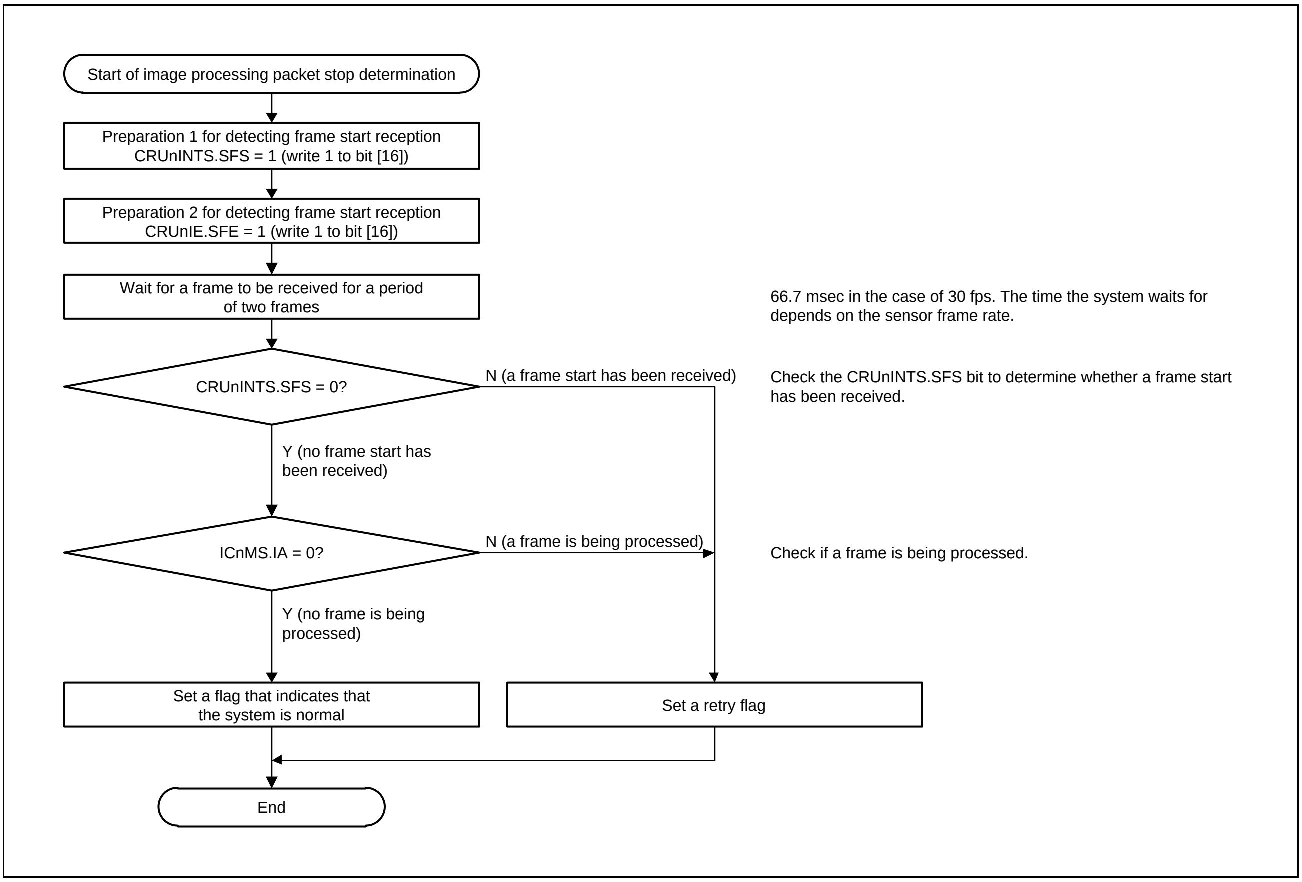
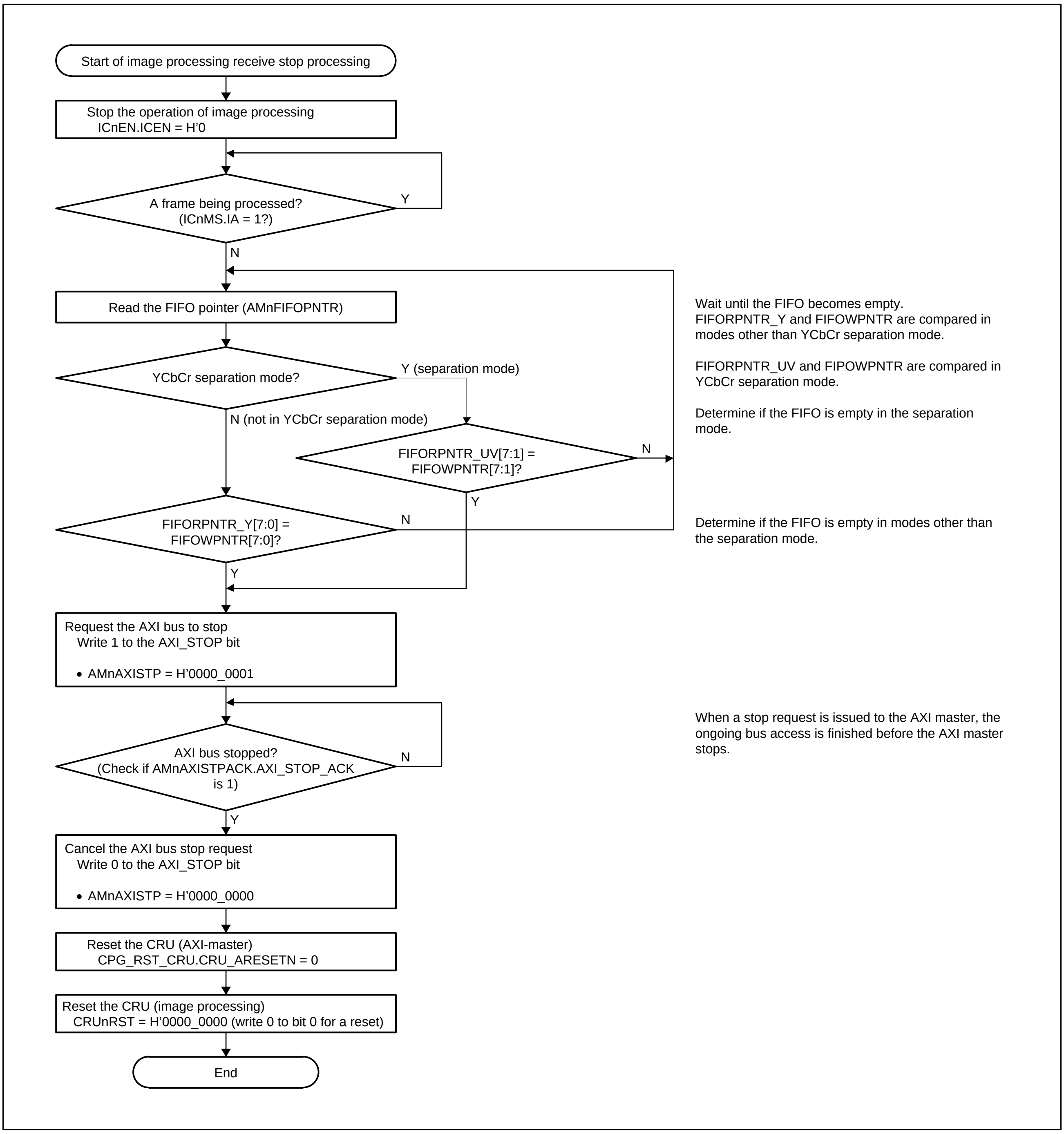


Figure 35.62 Determining Image Processing Reception Stop (Parallel Input)

35.3.5.2 Stopping Image Processing Reception (Parallel Input)



Wait until the FIFO becomes empty.
FIFORPNTR_Y and FIFOWPNTR are compared in modes other than YCbCr separation mode.

FIFORPNTR_UV and FIPOWPNTR are compared in YCbCr separation mode.

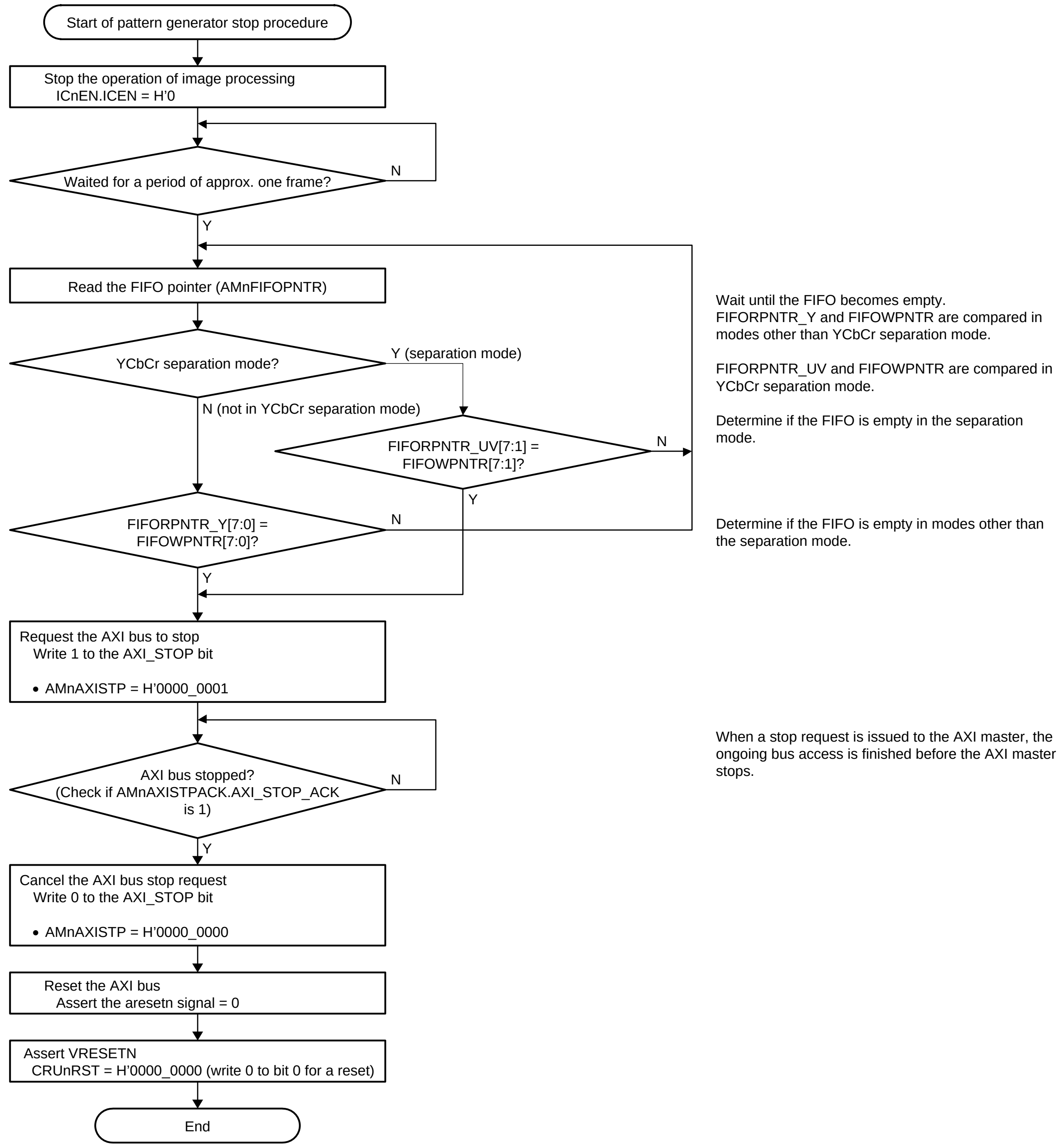
Determine if the FIFO is empty in the separation mode.

Determine if the FIFO is empty in modes other than the separation mode.

When a stop request is issued to the AXI master, the ongoing bus access is finished before the AXI master stops.

Figure 35.63 Stopping Image Processing Reception (Parallel Input)

35.3.6 Stopping the Pattern Generator



Wait until the FIFO becomes empty.
FIFORPNTR_Y and FIFOWPNTR are compared in modes other than YCbCr separation mode.

FIFORPNTR_UV and FIFOWPNTR are compared in YCbCr separation mode.

Determine if the FIFO is empty in the separation mode.

Determine if the FIFO is empty in modes other than the separation mode.

When a stop request is issued to the AXI master, the ongoing bus access is finished before the AXI master stops.

Note: The waiting time is determined by the pattern generator setting values.

Figure 35.64 Flow for Stopping the Pattern Generator

36. Image Scaling Unit (ISU)

The ISU is a module that reads out the image stored in the external DRAM, scale down image size, and outputs the reduced image into the DRAM. ISU also support a color format conversion and cropping image.

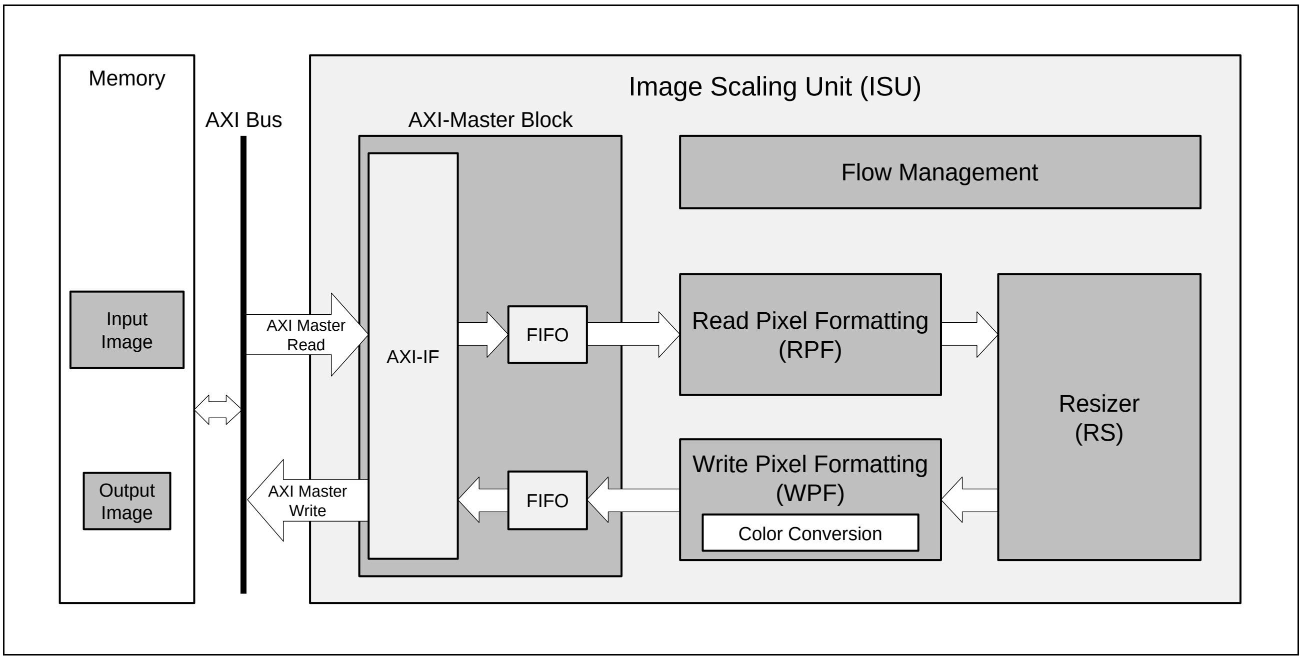


Figure 36.1 ISU Block Diagram

36.1 Features

The ISU functionality consists of “Resizer” and “Color Format Conversion”. “Color Format Conversion” can convert color space and color format (RGB/ARGB, YCbCr/YUV, RAW).

- Resizer
  - Scaling ratio: × 1/1 to 1/15 (Scale down)
  - Scaling algorithm: Bilinear
  - Horizontal and Vertical image scaling ratio are adjustable individually.
  - Cropping image

- Color Space and Color Format Conversion
  - Supported color format: RGB/ARGB, YCbCr/YUV, RAW(Grayscale)
    A(Alpha) of ARGB format is transparency. Zero is transparent (colorless).
  - Color space conversion by using 3 × 3 matrix which coefficients can be set arbitrary.
  - Convert color space and color format each other.
  - Endian Correction.

Table 36.1 Supported Image Format

Item	Description
Input	● Color Format – RGB/ARGB: 8 Formats RGB565, RGB888, BGR888, BGR666, ARGB8888, ARGB1555, RGBA8888, ABGR8888 – YCbCr/YUV: 4 Format YCbCr422 (Interleave) = UYVY YCbCr422 (Interleave) = YUY2 (YUYV) YCbCr422 (Semi-Planar) = NV16 YCbCr420 (Semi-Planar) = NV12 – RAW: 3 Format RAW8, RAW10, RAW12 (Grayscale)*[1] ● Maximum Size 2800 × 2047
Output	● Color Format – RGB/ARGB: 8 Format RGB565, RGB888, BGR888, BGR666, ARGB8888, ARGB1555, RGBA8888, ABGR8888 – YCbCr/YUV: 4 Format YCbCr422 (Interleave) = UYVY YCbCr422 (Interleave) = YUY2 (YUYV) YCbCr422 (Semi-Planar) = NV16 YCbCr420 (Semi-Planar) = NV12 – RAW: 3 Format RAW8, RAW10, RAW12 (Grayscale)*[1] ● Maximum Size 1920 × 1080

Note 1. RAW10 and RAW12 are rounded to 8-bit internally.

36.1.1 Flow Management (FM)

Controls the processing of the entire ISU. The frame processing is controlled by receiving the frame processing start instruction from the register. There are three types of frame processing:

1. Register Mode
2. Descriptor Mode (without automatic start of next frame)
3. Descriptor Mode (with automatic start of next frame)

36.1.1.1 Frame Processing Mode

There are the following three types of frame processing methods depending on the register settings.

- Register Setting Mode (1) is one-time processing operation according to register setting at once.
- Descriptor Mode without Auto-start (2) is also one-time processing operation according descriptor, not register setting. Descriptor Mode with Auto-start (3) automatically reads out the descriptor and continues its operation until all descriptors have been processed.

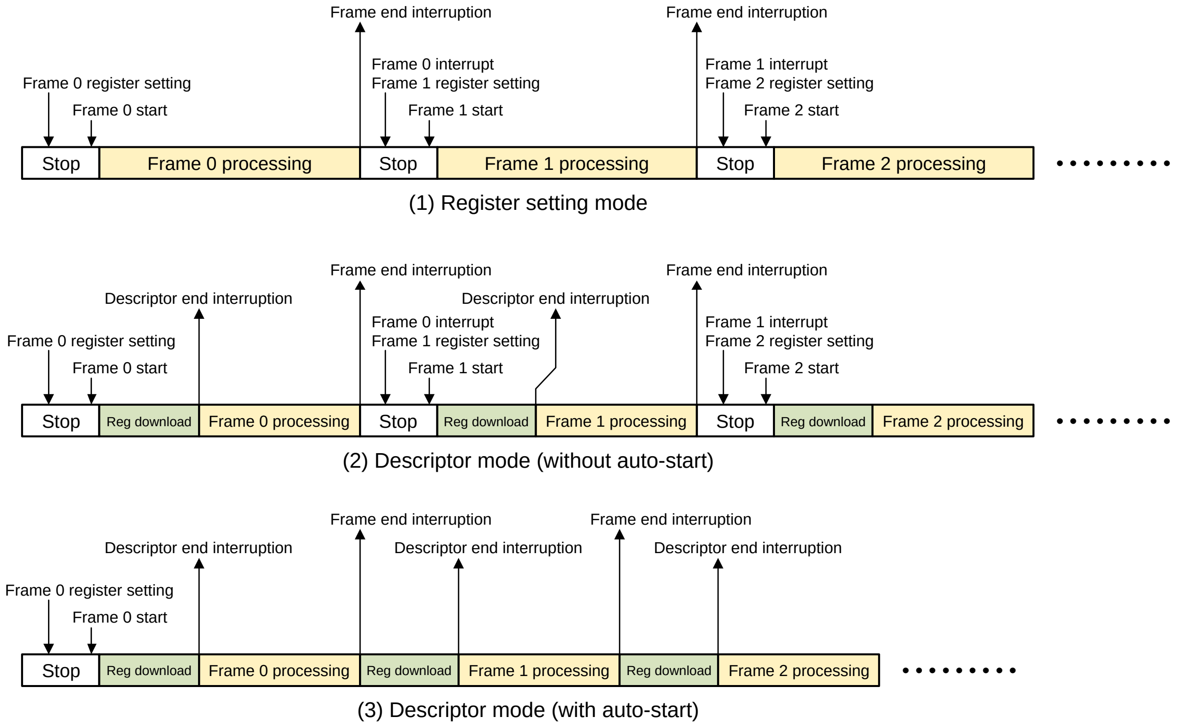


Figure 36.2 Frame Processing Mode

36.1.1.2 Descriptor

Descriptor Mode is the processing which reads out the descriptor format data stored in the memory and sets the registers automatically. It also includes the Auto-start setting for the next frame.

Flow Management (FM) read this descriptor setting data while not frame processing.

(1) Descriptor Format

The descriptor format is shown below.

Table 36.2 Descriptor Format

No.	Category	Item	Byte	Setting Range	Description
1	Header*1 *2	Descriptor Table Byte Count Setting	4 bytes	H’0000_0000 to H’0000_2000	Up to 8 Kbytes. The total number of bytes of register address and register setting data.
2	Register Setting Main body*3 *4 *5	Register Address 1	4 bytes	H’0000_0100 to H’0000_01F0	1st Register Setting Address
		Register Setting Data 1	4 bytes	H’0000_0000 to H’FFFF_FFFF	1st Register Setting Data
		Register Address 2	4 bytes	H’0000_0100 to H’0000_01F0	2nd Register Setting Address
		Register Setting Data 2	4 bytes	H’0000_0000 to H’FFFF_FFFF	2nd Register Setting Data
		Register Address 3	4 bytes	H’0000_0100 to H’0000_01F0	3rd Register Setting Address
		Register Setting Data 3	4 bytes	H’0000_0000 to H’FFFF_FFFF	3rd Register Setting Data
		Register Address 4	4 bytes	H’0000_0100 to H’0000_01F0	4th Register Setting Address
		Register Setting Data 4	4 bytes	H’0000_0000 to H’FFFF_FFFF	4th Register Setting Data
		•	•	•	•
		•	•	•	•
		•	•	•	•
3	Footer*2	Next Descriptor Table Address	8 bytes	H’0000_0000_0000_0000 to H’0000_0002_3FFF_FFFF	The next address is a multiple of 32 bytes. The setting range will be changed according to the product specifications.
		Control identifier register	4 bytes	H’0000_0000 to H’0000_0003	b31 to 2: All-0
					b1: Select bit to generate a frame end interruption. 0: Does not generate a Frame end interruption. 1: Generate a Frame end interruption.
					b0: Select bit for auto-start of next frame or end at this frame. 0: Disable next frame auto-start. (Exit without next frame operation.) 1: Enable next frame auto-start.

Note 1. The location of the Descriptor List is a 32-byte boundary. (Address that is a multiple of 32-byte) 0,32,64 ...

Note 2. The number of bytes is fixed for the header and footer. The register body is 8 bytes per set (address 4 bytes, data 4 bytes)

Note 3. The registers that can be set in the descriptor are limited to the Common register.

Note 4. Operation when the same register address is described in multiple addresses is not guaranteed.

Note 5. Specify in the offset address to be added to the Base Address(H’1084_0000). Specifically, the lower 10 bits [9: 0] should be the address of the register map, and the upper 22 bits [31:10] should be the address of All ‘0’. If the upper 22 bits are other than All ‘0’, it will be out of the register area and a list error will occur.

(2) Descriptor Error

An error flag is issued in the following cases.

(a) When the number of bytes in the header exceeds the limit, or when the value is not in units of 8 bytes

An error interrupt is generated, and subsequent frame processing is not executed.

Normal operation and recovery are not guaranteed unless the system is reset.

(b) When the register address is outside the specified address

A decoding error will occur if decoding is performed for items outside the register area.

Ignore the write to the register where the error occurred and continue the process itself.

36.1.2 AXI-Master

In response to a descriptor read request or input image read request from Flow Management (FM), the descriptor or input image is read from the DRAM. It also writes a reduced image to DRAM in response to a reduced image write request from WPF.

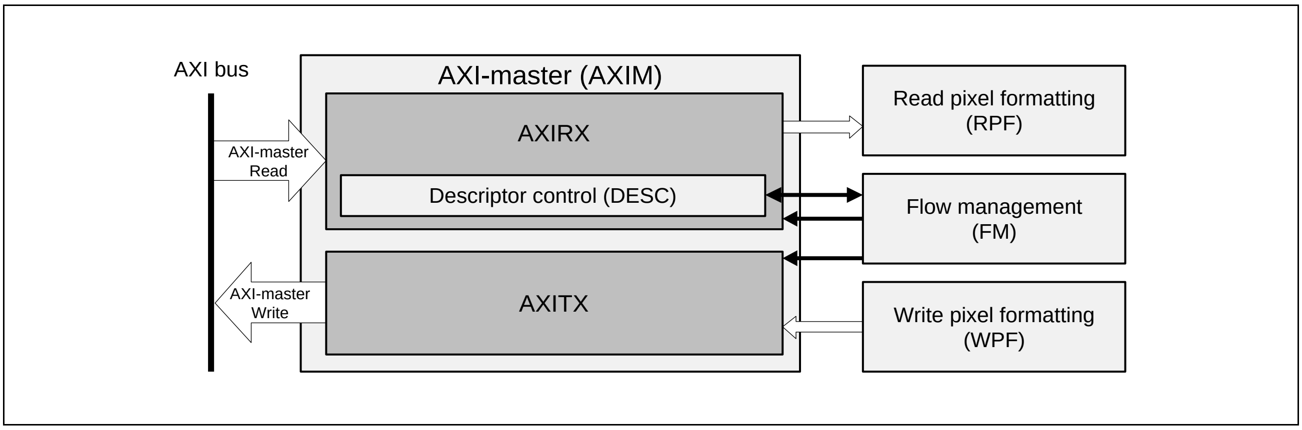


Figure 36.3 AXI-Master Block Diagram

36.1.2.1 Read Input Image (AXIRX)

In response to the descriptor read request from FM or the frame transfer start request, the descriptor or input image data is read from DRAM by AXI.

(1) Image Input Format

AXIRX support to read following formats from memory.

Table 36.3 Input Image Format

Read Method	Image Format	Variation
Interleave (Plane Number = 1)	ARGB (8 Types)	● The order of colors is different / the number of color bits is different. ● There may or may not be A.
	YCbCr422 8-bit (2 Types)	The two types differ only in the order of the colors of Cb and Cr.
	RAW (3 Types)	Grayscale 8-/10-/12-bit.
Semi-Planner (Plane Number = 2)	YCbCr422 8-bit (1 Types)	
	YCbCr420 8-bit (1 Types)	

- Interleave

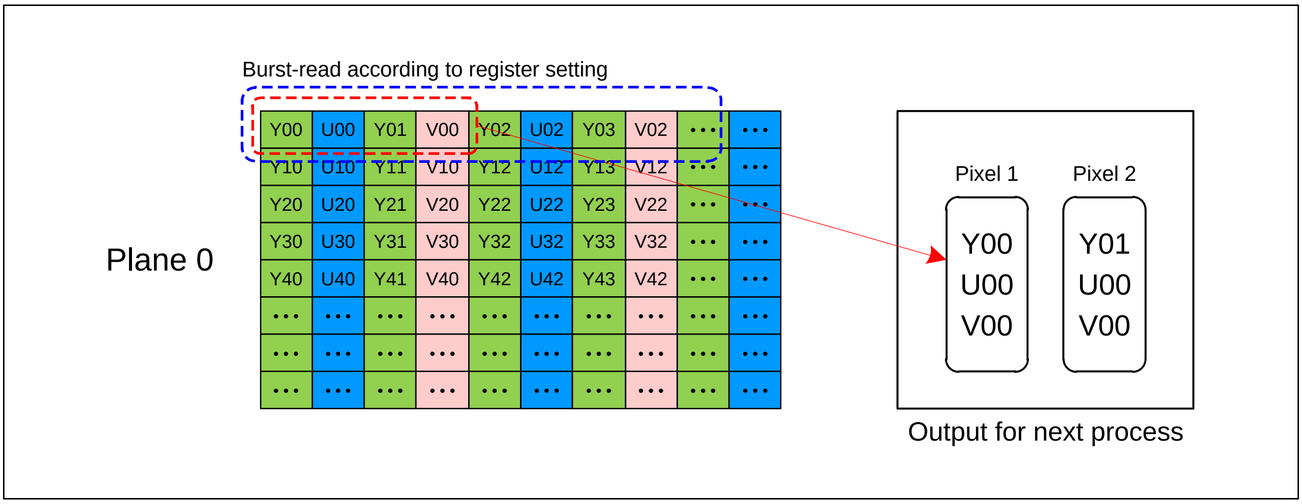


Figure 36.4 Example of Interleave Format for YCbCr422 (YUV422)

- Semi-planar

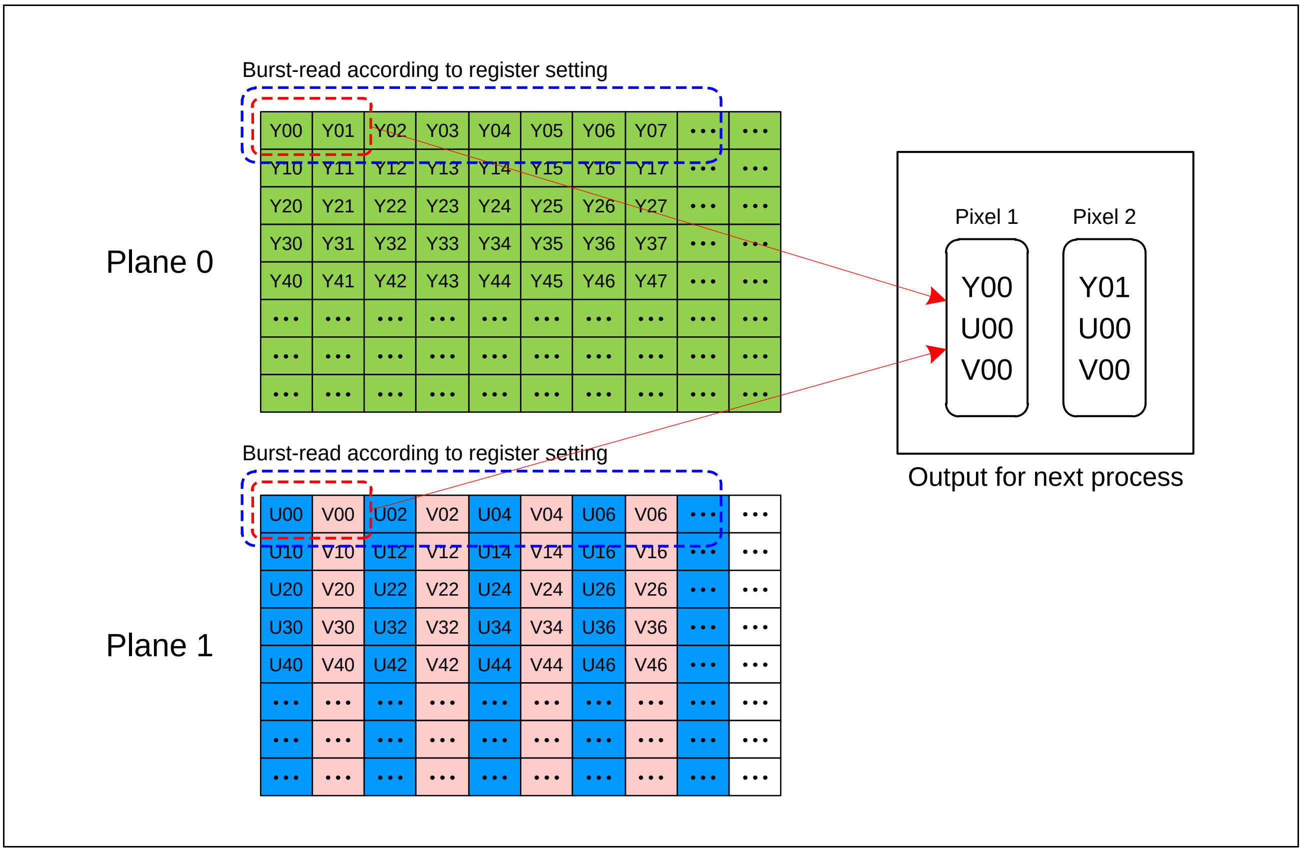


Figure 36.5 Example of Semi-planar Format for YCbCr422 (YUV422)

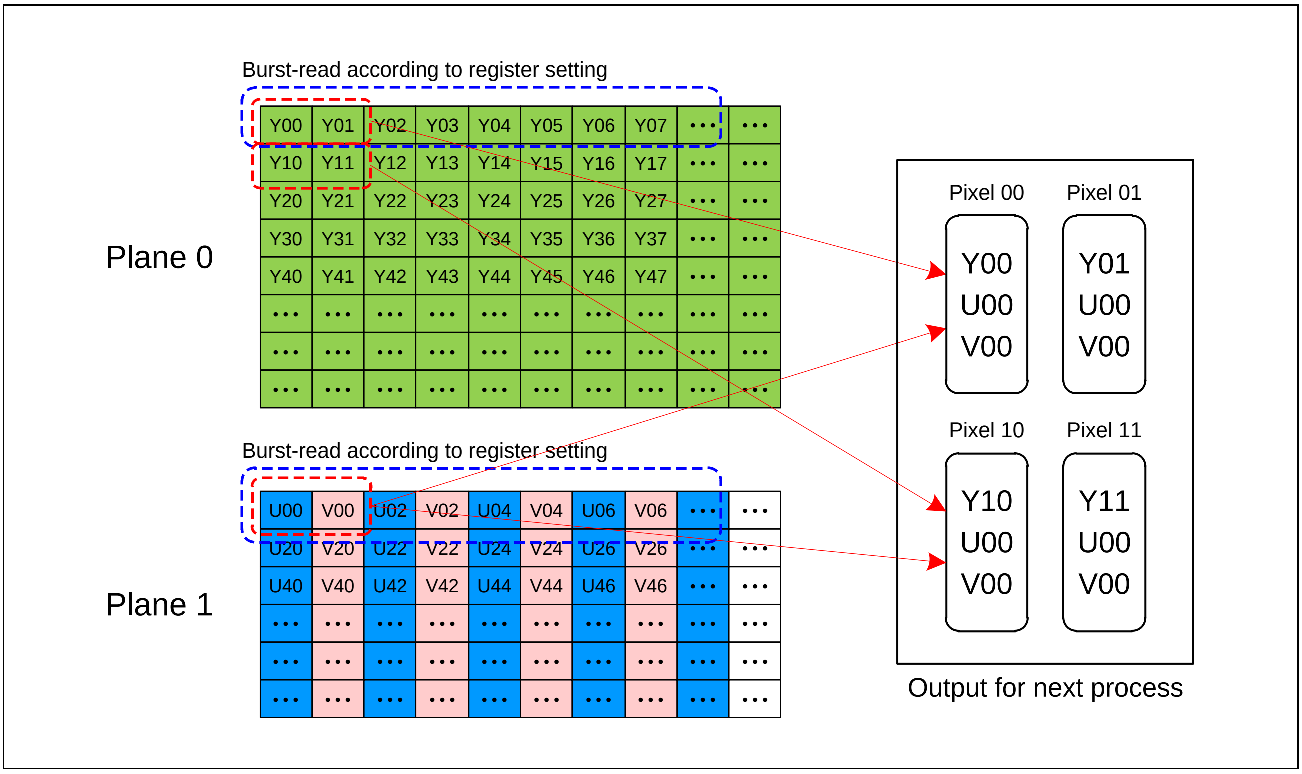


Figure 36.6 Example of Semi-planar format for YCbCr420 (YUV420)

(2) Input Setting

Data is read from each plane according to following register setting. Registers are required to set for each plane (Max 2 Plane).

Table 36.4 Input Image Setting

No.	Category	Register	Variation
(1)	Start Address	ISU_RPF_SRC_ADDH_PL0.SADD_PL0[1:0] ISU_RPF_SRC_ADDL_PL0.SADD_PL0[31:0]	● Start address for Plane0 of input image.
		ISU_RPF_SRC_ADDH_PL0.SADD_PL1[1:0] ISU_RPF_SRC_ADDL_PL0.SADD_PL1[31:0]	● Start address for Plane1 of input image.
(2)	Image Stride	ISU_RPF_SRC_STRD.SSTRPL0[15:0]	● Image stride of Plane0 [Byte]
		ISU_RPF_SRC_STRD.SSTRPL1[15:0]	● Image stride of Plane1 [Byte]
(3)	Input Area	ISU_RPF_SRC_SIZE.S_HSIZE[11:0]	● Horizontal size of processing area [Pixel] ● Max 2800 pixels
(4)		ISU_RPF_SRC_SIZE.S_VSIZE[10:0]	● Vertical size of processing area [Pixel] ● Max 2047 pixels

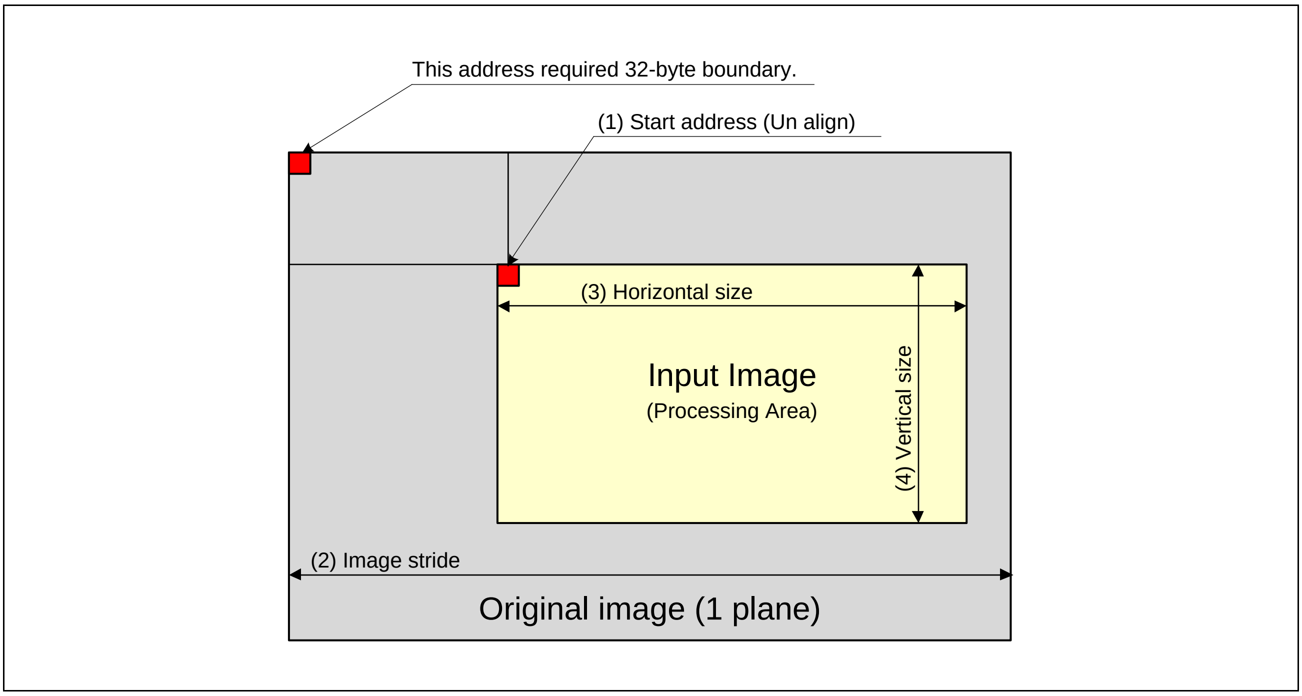


Figure 36.7 Input Area (Processing Area) Setting

(3) Write Output Image (AXITX)

This function write the resized (reduced) image data to DRAM.

(4) Support Format

AXITX support same format as AXIRX.

(5) Output Setting

Data is written out for each plane according to following setting. Registers are required to set for each plane (Max plane number is 2).

Table 36.5 Output Image Setting

No.	Category	Register	Variation
(1)	Start Address	ISU_WPF_DST_ADDH_PL0.DADD_PL0[1:0] ISU_WPF_DST_ADDL_PL0.DADD_PL0[31:0]	• Start address of Plane0 for output image.
		ISU_WPF_DST_ADDH_PL1.DADD_PL1[1:0] ISU_WPF_DST_ADDL_PL1.DADD_PL1[31:0]	• Start address of Plane1 for output image.
(2)	Image Stride	ISU_WPF_DST_STRD.DSTRPL0[15:0]	• Image stride of Plane0 [Byte]
		ISU_WPF_DST_STRD.DSTRPL1[15:0]	• Image stride of Plane1 [Byte]
(3)	Processing Area	ISU_RS_OS_CROP.O_HSIZE[10:0]	• Horizontal crop size [Pixel] • Max 1920 pixels
(4)		ISU_RS_OS_CROP.O_VSIZE[10:0]	• Vertical crop size [Pixel] • Max 1080 pixels

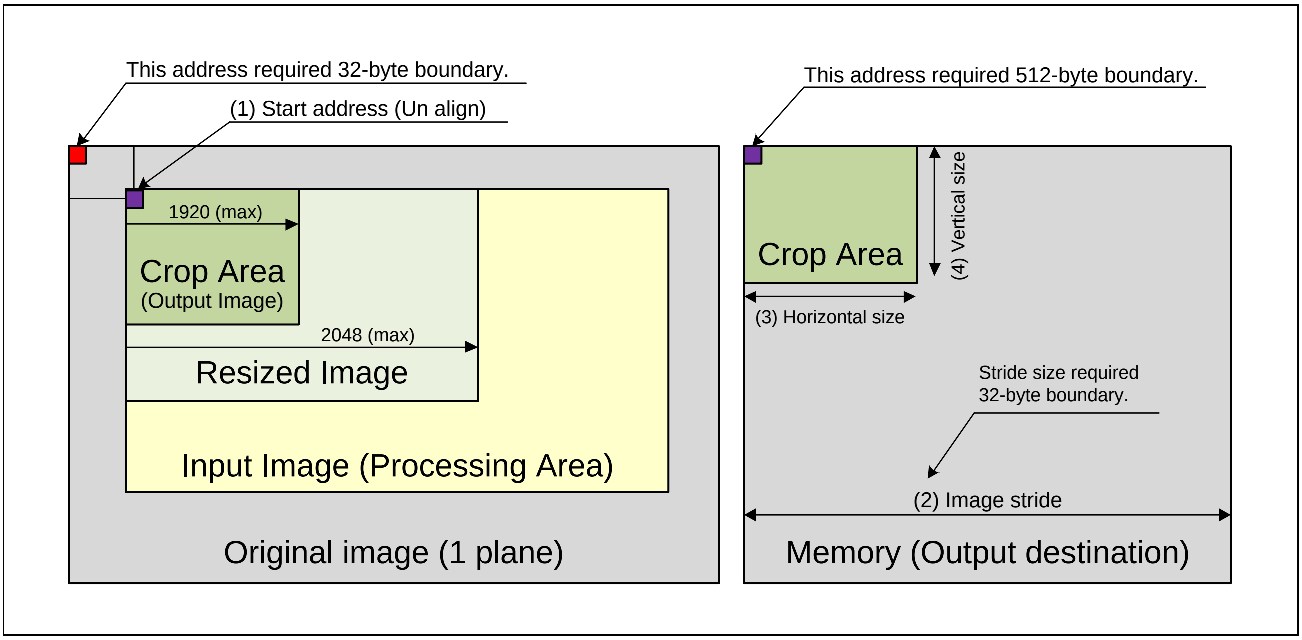


Figure 36.8 Output Setting

36.1.3 Read Pixel Formatting (RPF)

This function converts the input image data from AXI-Master and outputs it to the resizer. There are four types of conversion: Data swap, Data extraction, Offset binary conversion (YCbCr only), and pixel normalization. RPF also includes a test pattern generation function.

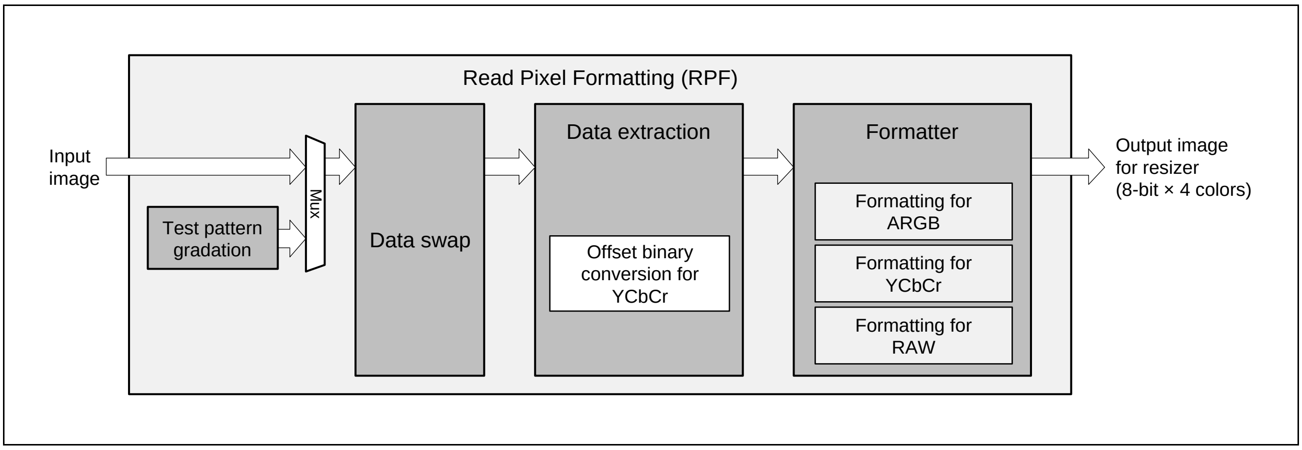
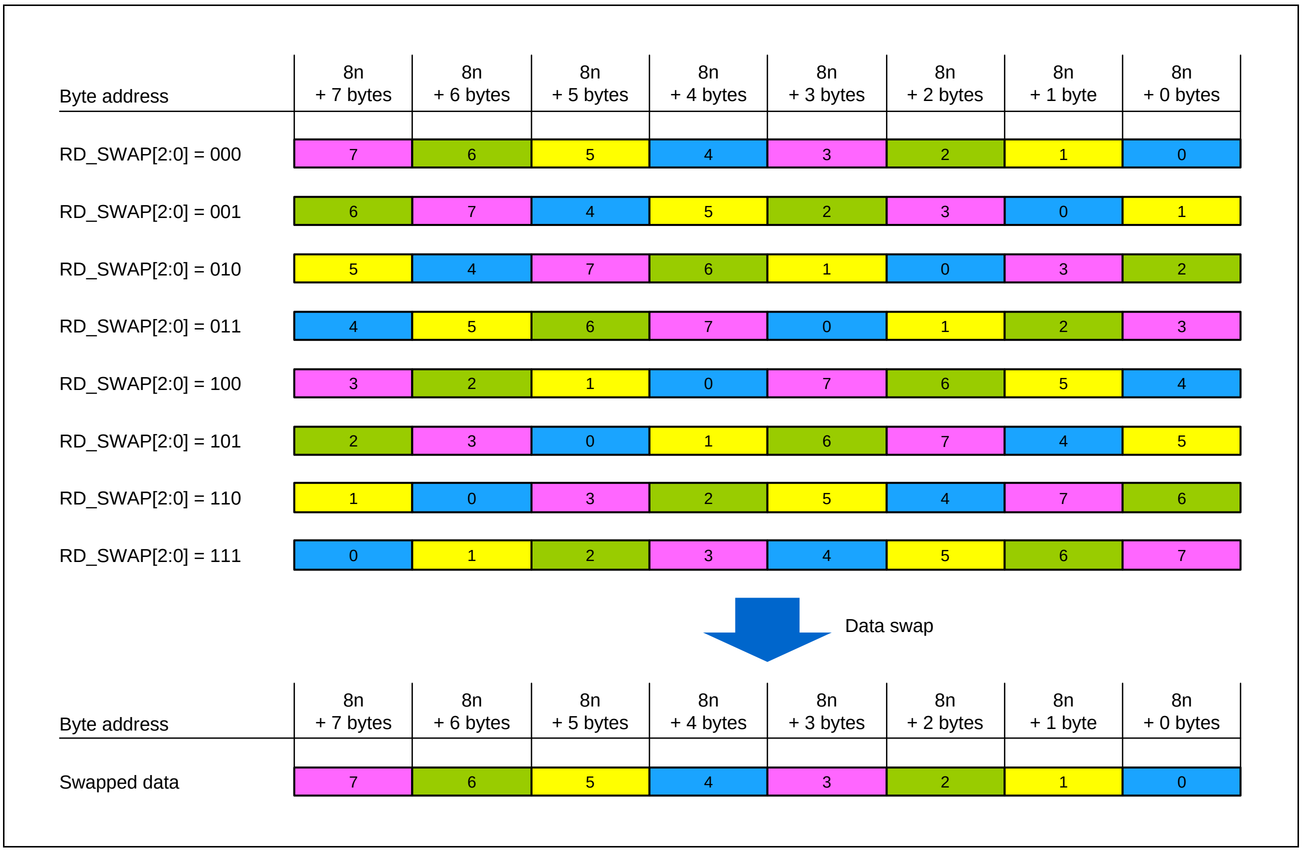


Figure 36.9 RPF Block Diagram

36.1.3.1 Data Swap (Endian Correction)

This function corrects byte order so that image data can be handled by ISU. Each bytes can be arranged any order according the register setting of ISU_RPF_SRC_DSWAP.RD_SWAP[2:0].



Byte address	8n + 7 bytes	8n + 6 bytes	8n + 5 bytes	8n + 4 bytes	8n + 3 bytes	8n + 2 bytes	8n + 1 byte	8n + 0 bytes
RD_SWAP[2:0] = 000	7	6	5	4	3	2	1	0
RD_SWAP[2:0] = 001	6	7	4	5	2	3	0	1
RD_SWAP[2:0] = 010	5	4	7	6	1	0	3	2
RD_SWAP[2:0] = 011	4	5	6	7	0	1	2	3
RD_SWAP[2:0] = 100	3	2	1	0	7	6	5	4
RD_SWAP[2:0] = 101	2	3	0	1	6	7	4	5
RD_SWAP[2:0] = 110	1	0	3	2	5	4	7	6
RD_SWAP[2:0] = 111	0	1	2	3	4	5	6	7

Data swap

Byte address	8n + 7 bytes	8n + 6 bytes	8n + 5 bytes	8n + 4 bytes	8n + 3 bytes	8n + 2 bytes	8n + 1 byte	8n + 0 bytes
Swapped data	7	6	5	4	3	2	1	0

Figure 36.10 Data Swap Correction

36.1.3.2 Formatter

8-bit normalization is required for ARGB/RGB and RAW formats since Resizer can handle 8-bit format only.

On the other hand, although YCbCr/YUV does not need 8-bit normalization since its components are already 8-bit, it is necessary to exchange into YCbCr/YUV444 format.

■ RGB/ARGB

This function support 8 ARGB/RGB formats as bellow. All formats are “Number of planes = 1”.

Table 36.6 ARGB Support Format

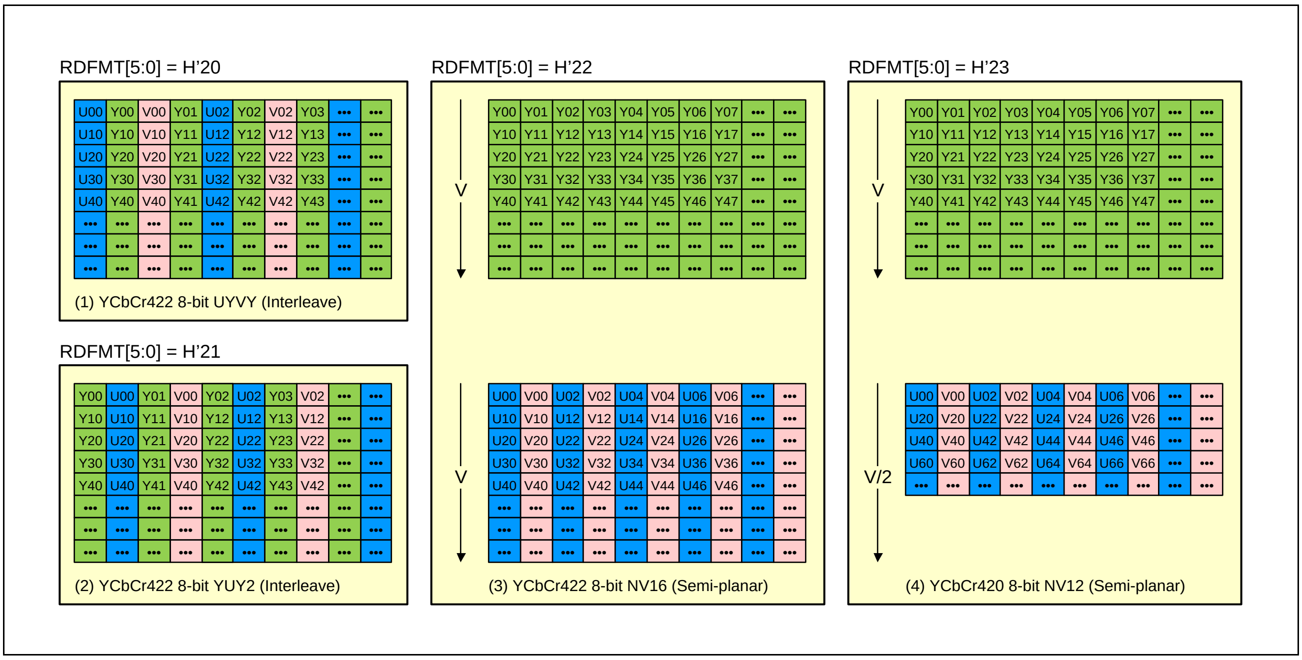
	Format				n								n + 1								n + 2								n + 3							
RDFMT [5:0]	Color Format	Padding	bit/pixel	phase	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
H'01	RGB565	—	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'05	ARGB8888	—	24	—	A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
H'06	RGBA8888	—		—	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
H'03	RGB888	—		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
				1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
				2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
H'04	BGR888	—	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
				1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
				2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
H'00	ARGB1555	—	15	—	A	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H'02	BGR666	BGR upper	18	0	▨	▨	B0	B0	B0	B0	B0	B0	▨	▨	G0	G0	G0	G0	G0	G0	▨	▨	R0	R0	R0	R0	R0	R0	▨	▨	B1	B1	B1	B1	B1	B1
				1	▨	▨	G1	G1	G1	G1	G1	G1	▨	▨	R1	R1	R1	R1	R1	R1	▨	▨	B2	B2	B2	B2	B2	B2	▨	▨	G2	G2	G2	G2	G2	G2
				2	▨	▨	R2	R2	R2	R2	R2	R2	▨	▨	B3	B3	B3	B3	B3	B3	▨	▨	G3	G3	G3	G3	G3	G3	▨	▨	R3	R3	R3	R3	R3	R3
H'07	ABGR8888	—	24	—	A	A	A	A	A	A	A	A	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0

■ YCbCr/YUV

This function support 4 YCrCb/YUV formats as bellow.

Table 36.7 YCbCr/YUV Support Format

Data Format	RDFMT[5:0]	Color Format	Alias	Figure Index
Interleave (Number of Planes = 1)	H’20	YCbCr422 8-bit	UYVY	
	H’21	YCbCr422 8-bit	YUY2	
Semi-Planar (Number of Planes = 2)	H’22	YCbCr422 8-bit	NV16	
	H’23	YCbCr420 8-bit	NV12	



(1) YCbCr422 8-bit UYVY (Interleave)

(2) YCbCr422 8-bit YUY2 (Interleave)

(3) YCbCr422 8-bit NV16 (Semi-planar)

(4) YCbCr420 8-bit NV12 (Semi-planar)

Figure 36.11 YCbCr/YUV Support Format

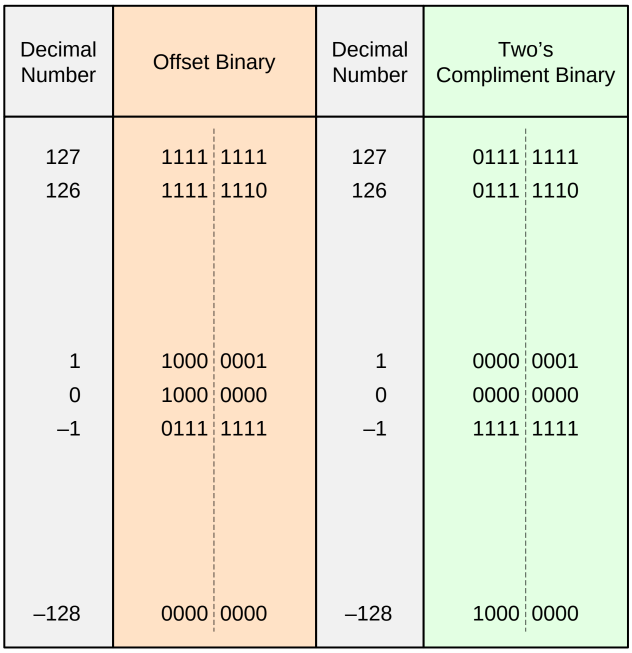
ISU can handle only Offset Binary as the chroma data format (CbCr/UV). Therefore, it is necessary to convert from Two’s Complement Binary to Offset Binary.

■ ISU_RPF_UVBIN.UVCHG = 0 (Default)

ISU uses original data on DRAM as chroma value when RDFMT = H’20 to H’23.
(CrCb/UV is Offset Binary.)

■ ISU_RPF_UVBIN.UVCHG = 1

ISU convert chroma data (CrCb/UV) from Two’s Complement Binary to Offset Binary.



Decimal Number	Offset Binary	Decimal Number	Two’s Compliment Binary
127	1111 1111	127	0111 1111
126	1111 1110	126	0111 1110
1	1000 0001	1	0000 0001
0	1000 0000	0	0000 0000
–1	0111 1111	–1	1111 1111
–128	0000 0000	–128	1000 0000

Figure 36.12 Types of Binary Expression

■ RAW

This function support 3 RAW formats as bellow. All formats are “Number of planes = 1”.

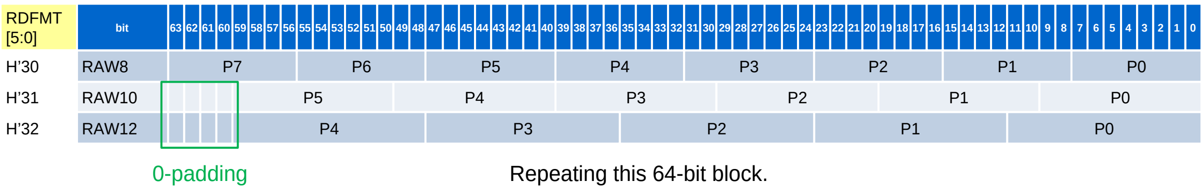


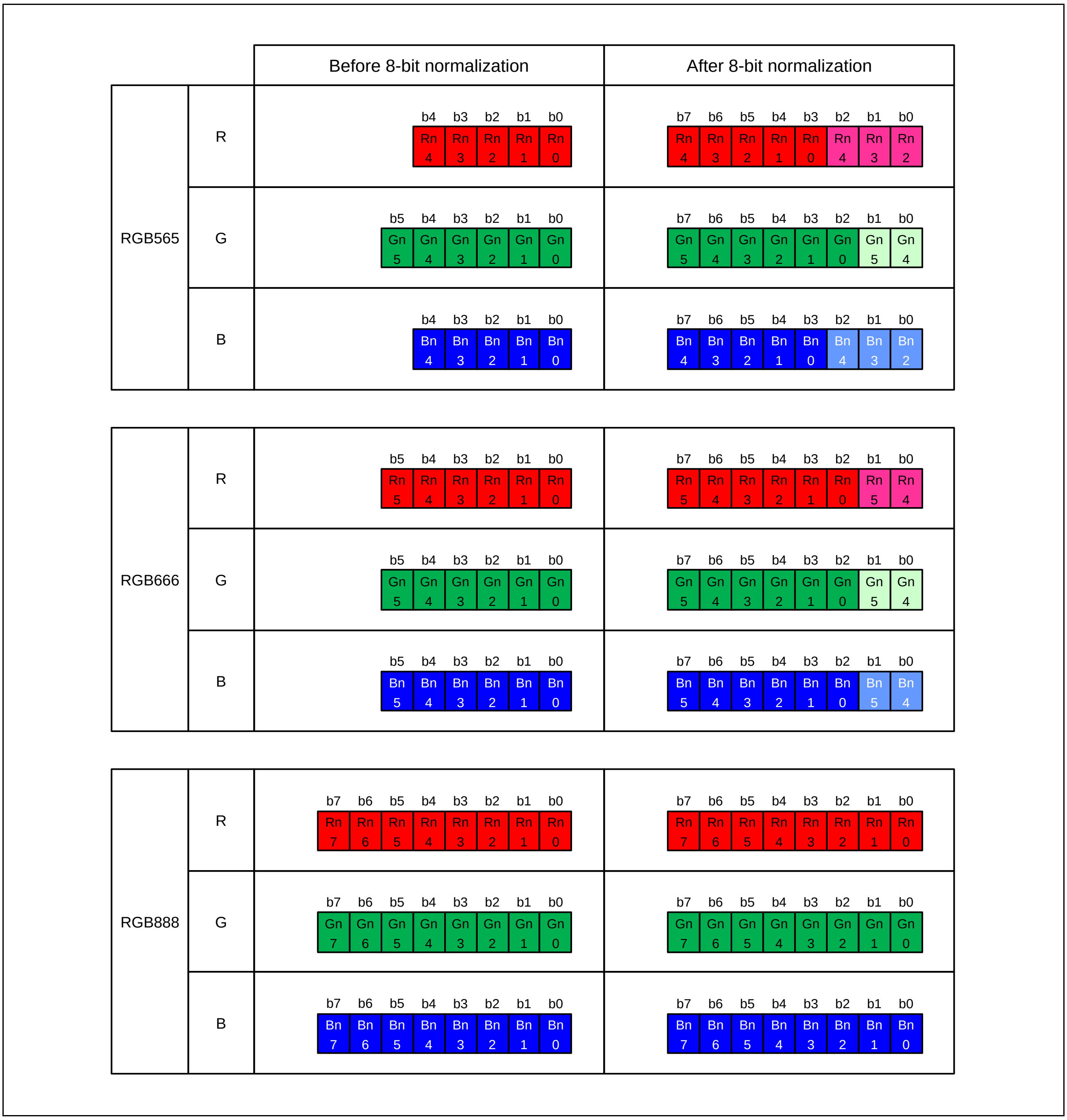
Figure 36.13 RAW Support Format

36.1.3.3 Data Extraction

These functions extract each color components (ex. R, G, B) and normalized as 8-bit for each so that image data can be handled by Resizer. Appropriate processing is performed for each input color format (RGB/ARGB, YCbCr/YUV, RAW) by setting of ISU_RPF_FMT.RDFMT[5:0].

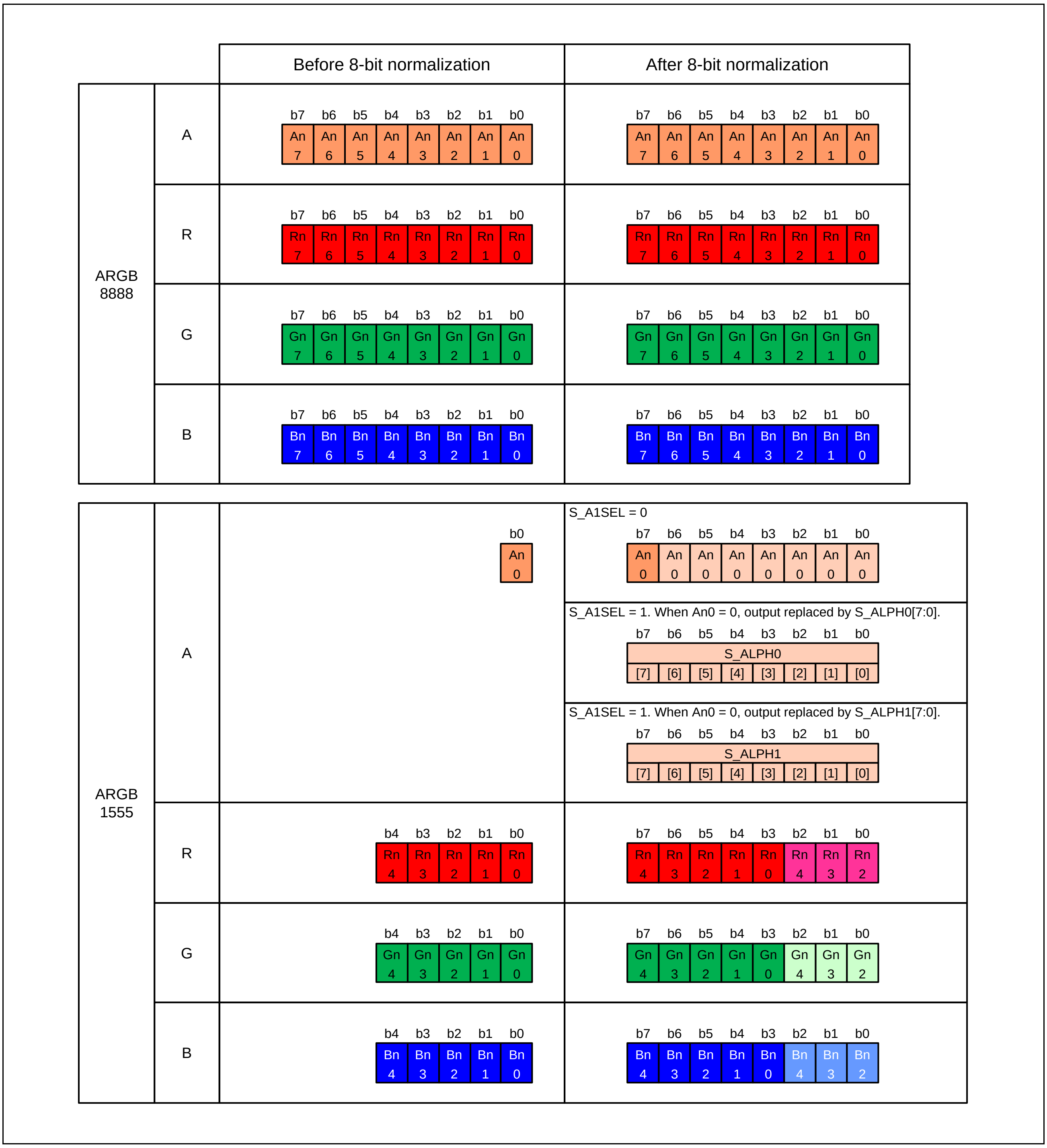
■ RGB/ARGB

For RGB/ARGB format normalization, upper bits are copied to lower bits.



		Before 8-bit normalization	After 8-bit normalization
RGB565	R	b4 b3 b2 b1 b0 Rn4 Rn3 Rn2 Rn1 Rn0	b7 b6 b5 b4 b3 b2 b1 b0 Rn4 Rn3 Rn2 Rn1 Rn0 Rn4 Rn3 Rn2
	G	b5 b4 b3 b2 b1 b0 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0	b7 b6 b5 b4 b3 b2 b1 b0 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0 Gn5 Gn4
	B	b4 b3 b2 b1 b0 Bn4 Bn3 Bn2 Bn1 Bn0	b7 b6 b5 b4 b3 b2 b1 b0 Bn4 Bn3 Bn2 Bn1 Bn0 Bn4 Bn3 Bn2
RGB666	R	b5 b4 b3 b2 b1 b0 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0	b7 b6 b5 b4 b3 b2 b1 b0 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0 Rn5 Rn4
	G	b5 b4 b3 b2 b1 b0 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0	b7 b6 b5 b4 b3 b2 b1 b0 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0 Gn5 Gn4
	B	b5 b4 b3 b2 b1 b0 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0	b7 b6 b5 b4 b3 b2 b1 b0 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0 Bn5 Bn4
RGB888	R	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0
	G	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0
	B	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0

Figure 36.14 8-bit Normalization for RGB Format



		Before 8-bit normalization	After 8-bit normalization
ARGB 8888	A	b7 b6 b5 b4 b3 b2 b1 b0 An7 An6 An5 An4 An3 An2 An1 An0	b7 b6 b5 b4 b3 b2 b1 b0 An7 An6 An5 An4 An3 An2 An1 An0
	R	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0
	G	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0
	B	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0
ARGB 1555	A	b0 An0	S_A1SEL = 0 b7 b6 b5 b4 b3 b2 b1 b0 An0 An0 An0 An0 An0 An0 An0 An0 S_A1SEL = 1. When An0 = 0, output replaced by S_ALPH0[7:0]. b7 b6 b5 b4 b3 b2 b1 b0 S_ALPH0 [7] [6] [5] [4] [3] [2] [1] [0] S_A1SEL = 1. When An0 = 0, output replaced by S_ALPH1[7:0]. b7 b6 b5 b4 b3 b2 b1 b0 S_ALPH1 [7] [6] [5] [4] [3] [2] [1] [0]
	R	b4 b3 b2 b1 b0 Rn4 Rn3 Rn2 Rn1 Rn0	b7 b6 b5 b4 b3 b2 b1 b0 Rn4 Rn3 Rn2 Rn1 Rn0 Rn4 Rn3 Rn2
	G	b4 b3 b2 b1 b0 Gn4 Gn3 Gn2 Gn1 Gn0	b7 b6 b5 b4 b3 b2 b1 b0 Gn4 Gn3 Gn2 Gn1 Gn0 Gn4 Gn3 Gn2
	B	b4 b3 b2 b1 b0 Bn4 Bn3 Bn2 Bn1 Bn0	b7 b6 b5 b4 b3 b2 b1 b0 Bn4 Bn3 Bn2 Bn1 Bn0 Bn4 Bn3 Bn2

Figure 36.15 8-bit Normalization for ARGB Format

■ YCbCr/YUV

For YCbCr/YUV420 Semi-Planar format, even line re-use previous odd line’s chroma information.

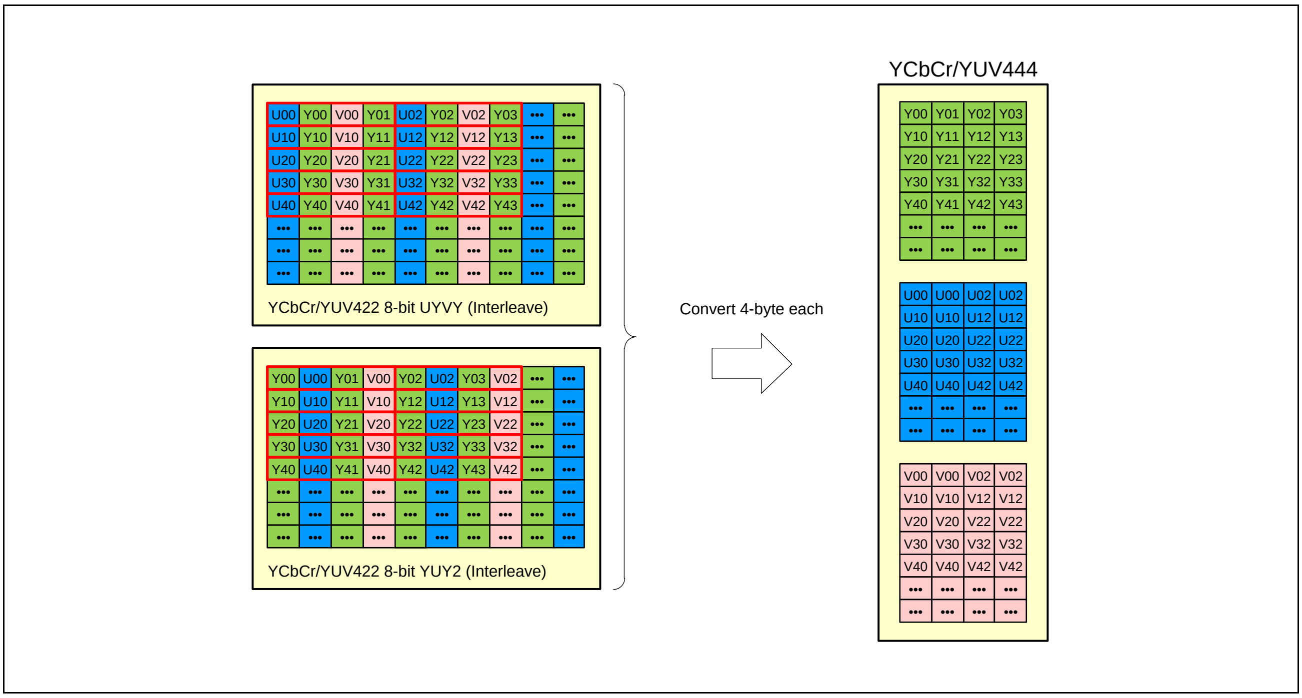


Figure 36.16 YCbCr/YUV444 Conversion from YCbCr/YUV422 Format

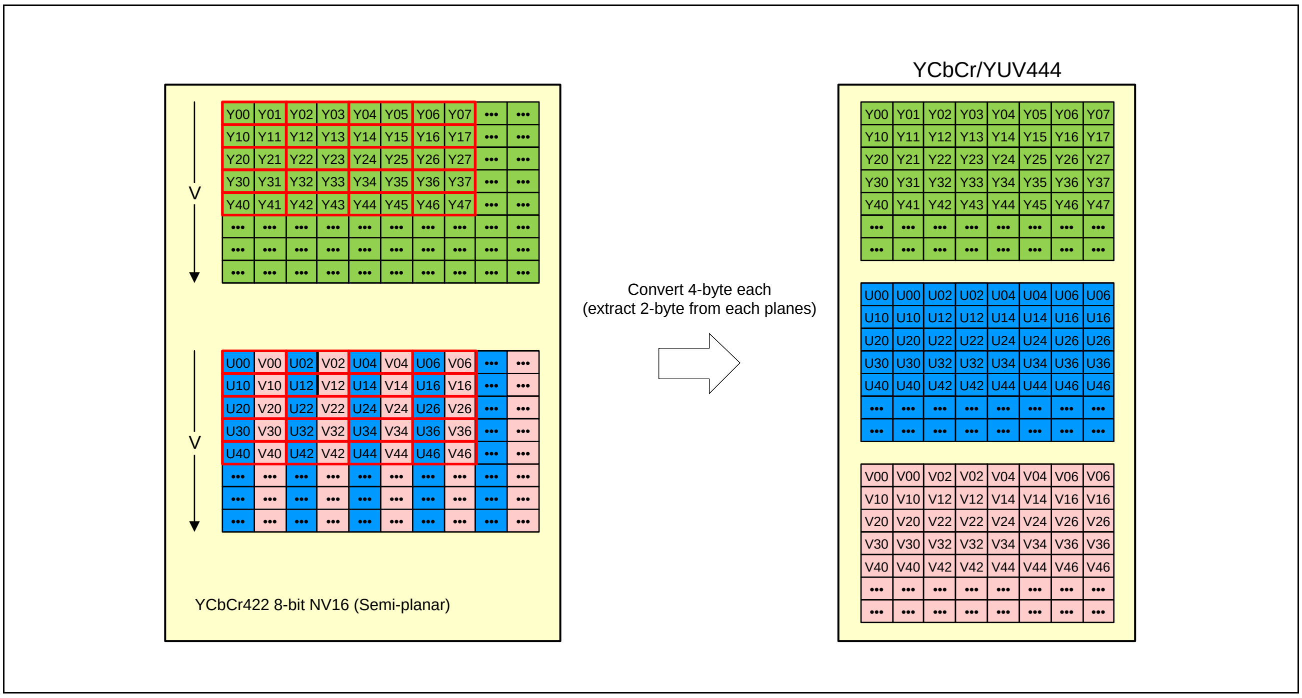


Figure 36.17 YCbCr/YUV444 Conversion from YCbCr/YUV422 Semi-Planar Format

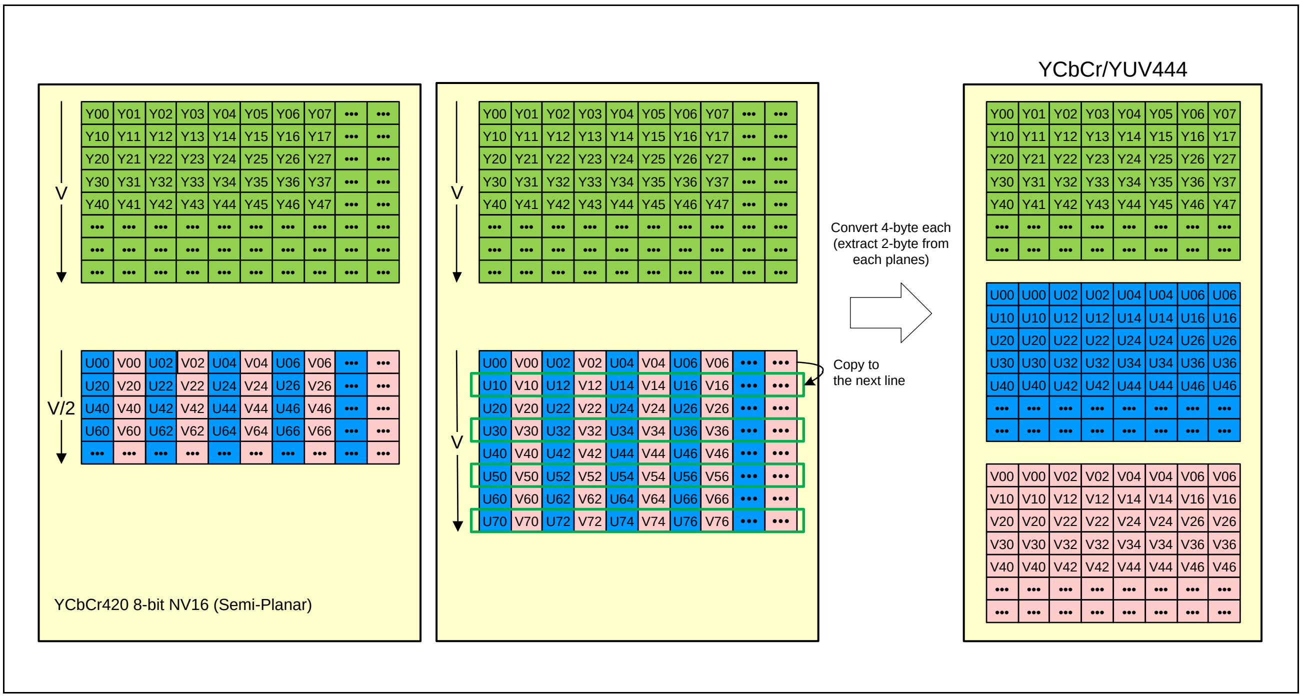


Figure 36.18 YCbCr/YUV444 Conversion from YCbCr/YUV420 Semi-Planar Format

■ RAW

RAW is converted to 8-bit by using upper 8-bit of RAW8-/10-/12-bit.

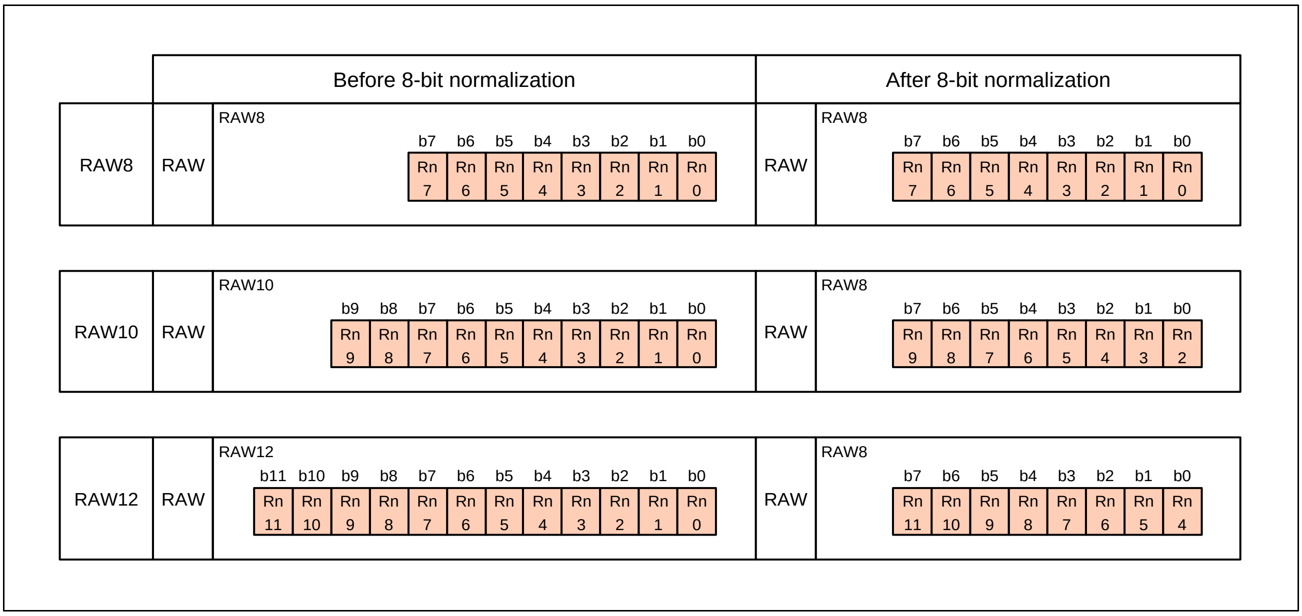


Figure 36.19 8-bit Normalization for RAW Format

36.1.3.4 Test Pattern Generation

Instead of the input image data from AXI-Master, RPF can generates a test pattern and outputs it to the Resizer.

The test pattern is an 8-bit gradation image from the upper left to the lower right in ARGB8888 format according to the register setting.

Please switch between the test pattern and the input image data from AXI-Master according below procedure.

1. Enable the test pattern. (ISU_RPF_SRC_TD1.TMODE_EN = 1b)
2. Start the resizing process. (ISU_FM_FRCON.START = 1b)

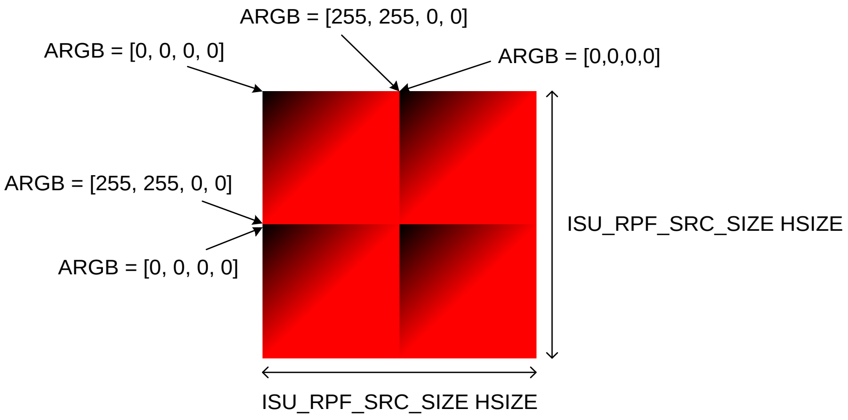


Figure 36.20 Test Pattern of R Gradation

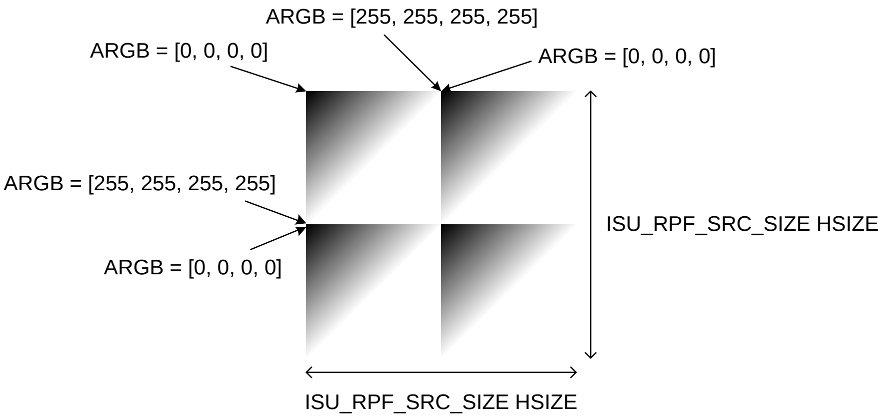


Figure 36.21 Pattern of R,G,B Gradation

36.1.4 Resizer (RS)

Resizer (RS) reduces the image size by using bi-linear algorithm (4-point linear interpolation) according to register setting of horizontal / vertical resize ratio.

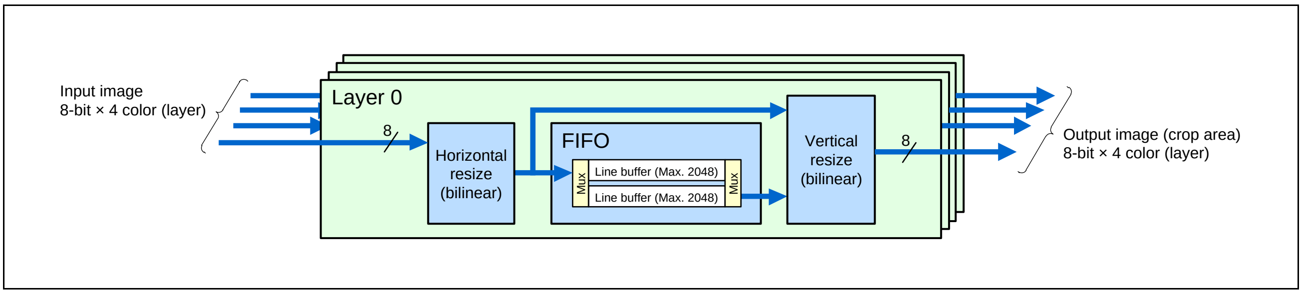


Figure 36.22 Resizer Block Diagram

The resize processing is different whether the Resized image size is smaller or larger than the Crop area.

If the Resized image size is smaller than the Crop area, fill in the missing edge pixels according to the register settings below.

When ISU_RS_PADDMODE.PADSEL = 0, Fill in the padding area by copying the right / bottom edge pixels after the reduction.

When ISU_RS_PADDMODE.PADSEL = 1, Fill in the padding area with the value specified by the register.

If the Resized image size is larger than the Crop area, it will be cut off by the Crop area.

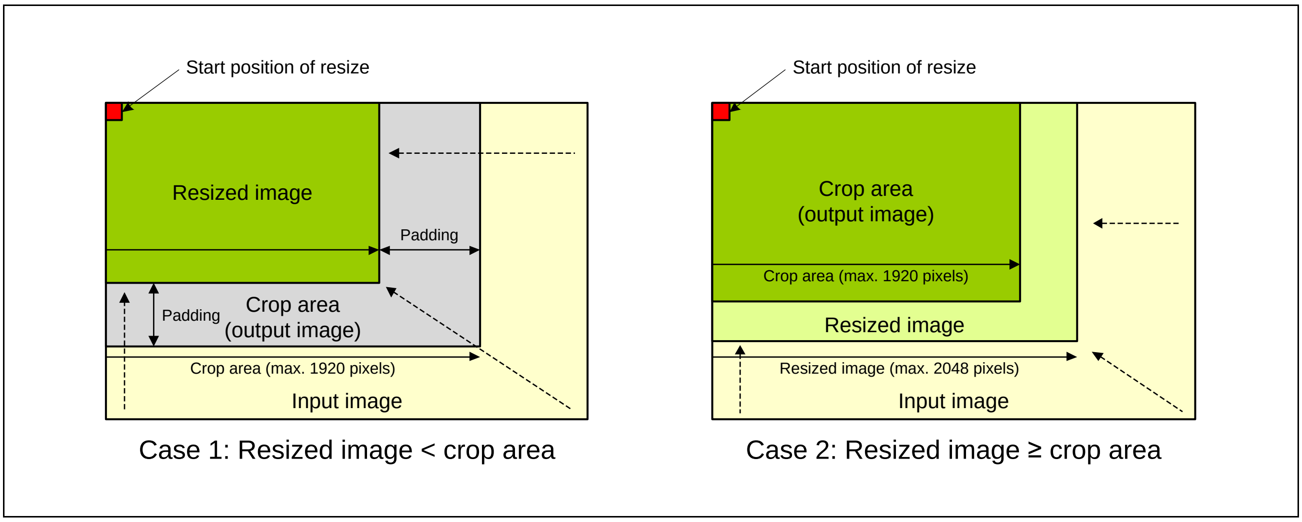


Figure 36.23 Relationship of Resized Image and Crop Area

36.1.4.1 Resize Start Position

The resize processing start from Resize Start Position (ISU_RS_STPOS) on the input image in according to the register setting. This position setting is specified in pixel coordinates from the top left of the image. If 0 is specified, the input image will be the Resizer (RS) target image as it is.

Padding area will be filled based on the setting of ISU_RS_PADDMODE.PADSEL.

- Register ISU_RS_STPOS.HSTART = Resize Start H position
- Register ISU_RS_STPOS.VSTART = Resize Start V position

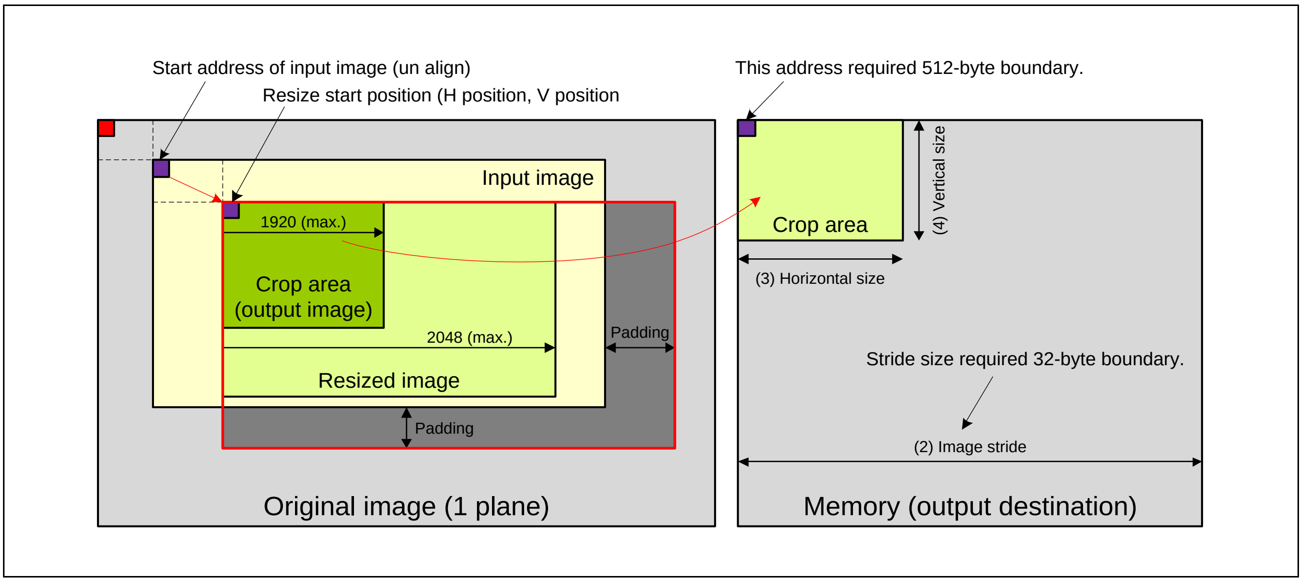


Figure 36.24 Resize Start Position

36.1.4.2 Sub Pixel Offset

The Resize Start Position Tuning (ISU_RS_POS_TUNE) can be used to adjust the start position by sub pixel level to reduce moiré.

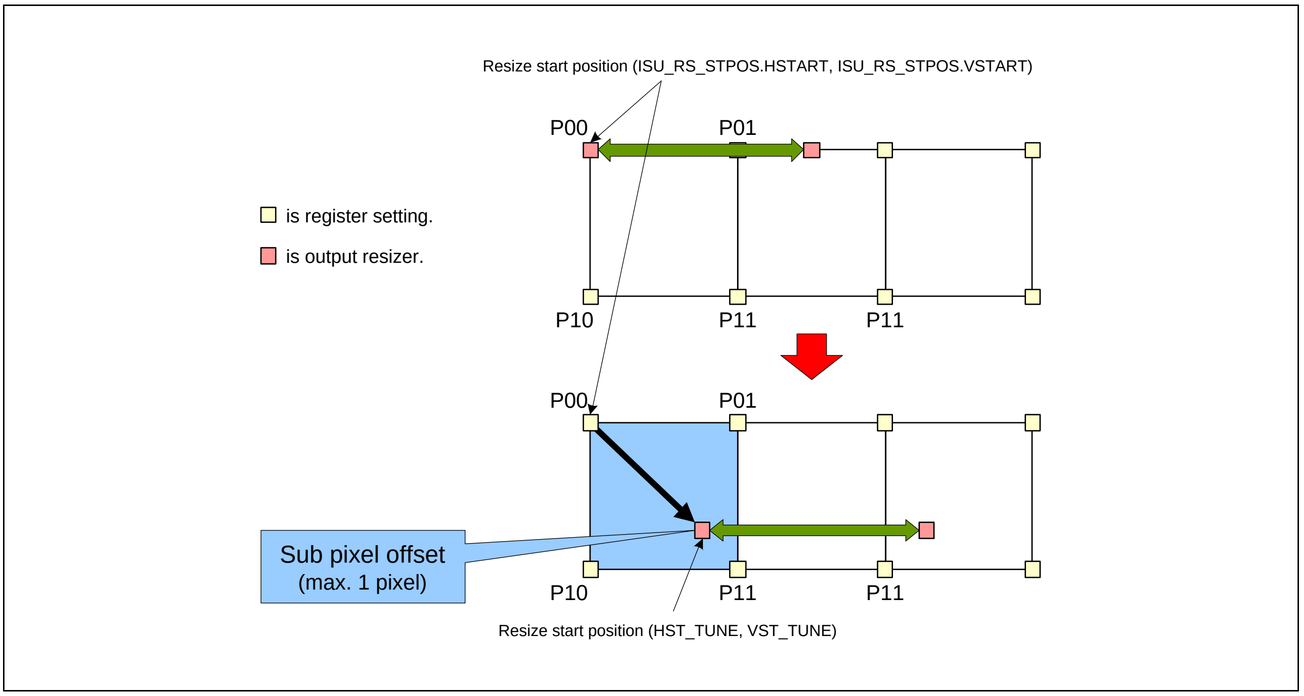


Figure 36.25 Sub Pixel Position Adjustment

36.1.4.3 Scaling Calculation

The resizer can be handled resizing magnification independently for horizontally and vertically. The image size of this processing result can be calculated approximately by using “Input image horizontal size x hscale” and “Input image vertical size × vscale”.

Table 36.8 Scaling Coefficient Setting

No.	Category	Register	Variation
(1)	Horizontal Scaling Ratio	RS_HSCALE.HMANT[3:0]	- m_h: Integer part of Horizontal scaling coefficient.
		RS_HSCALE.HFRAC[11:0]	- f_h: Fraction part of Horizontal scaling coefficient.
(2)	Vertical Scaling Ratio	RS_VSCALE.VMANT[3:0]	- m_v: Integer part of Vertical scaling coefficient.
		RS_VSCALE.VFRAC[11:0]	- f_v: Fraction part of Vertical scaling coefficient.

$$\mathrm{hascale} = \frac{4096}{4096 \times \mathrm{m_h} + \mathrm{f_h}} \qquad \mathrm{vscale} = \frac{4096}{4096 \times \mathrm{m_v} + \mathrm{f_v}}$$

Figure 36.26 Sub Pixel Position Adjustment

$$\mathrm{hascale} = \frac{4096}{4096 \times \mathrm{m_h} + \mathrm{f_h}}$$

$\mathrm{hsize_{org}}$: Horizontal Size of Input Image

$$\mathrm{hsize_{down\_scaled}} = \langle 1 + \left[\langle 1 + \frac{\langle \mathrm{hsize_{org}} - 1 \rangle}{\mathrm{m}_{\acute{\mathrm{h}}}} \rangle - 1 \right] \times \mathrm{m}_{\acute{\mathrm{h}}} \times \mathrm{hscale} \rangle$$

$$\mathrm{hsize_{down\_scaled}} = \langle 1 + \left[\left[\langle 1 + \frac{\langle \mathrm{hsize_{org}} - 1 \rangle}{\mathrm{m}_{\acute{\mathrm{h}}}} \rangle - 1 \right] \times \mathrm{m}_{\acute{\mathrm{h}}} \times 4096 \right] / (4096 \times \mathrm{m_h} + \mathrm{f_h}) \rangle$$

Figure 36.27 Horizontal Scaling Calculation

$$\mathrm{vscale} = \frac{4096}{4096 \times \mathrm{m_v} + \mathrm{f_v}}$$

$\mathrm{vsize}_{\mathrm{org}}$: Vertcal Size of Input Image

$$\mathrm{vsize}_{\mathrm{down\_scaled}} = \langle 1 + \left[\langle 1 + \frac{\langle \mathrm{vsize}_{\mathrm{org}} - 1 \rangle}{\mathrm{m}_{\acute{\mathrm{v}}}} \rangle - 1 \right] \times \mathrm{m}_{\acute{\mathrm{v}}} \times \mathrm{vscale} \rangle$$

$$\mathrm{vsize}_{\mathrm{down\_scaled}} = \langle 1 + \left[\left[\langle 1 + \frac{\langle \mathrm{vsize}_{\mathrm{org}} - 1 \rangle}{\mathrm{m}_{\acute{\mathrm{v}}}} \rangle - 1 \right] \times \mathrm{m}_{\acute{\mathrm{v}}} \times 4096 \right] / (4096 \times \mathrm{m_v} + \mathrm{f_v}) \rangle$$

Figure 36.28 Vertical Scaling Calculation

36.1.4.4 Resize Processing

Following figure shows the processing image of bi-linear algorithm (4-point interpolation).

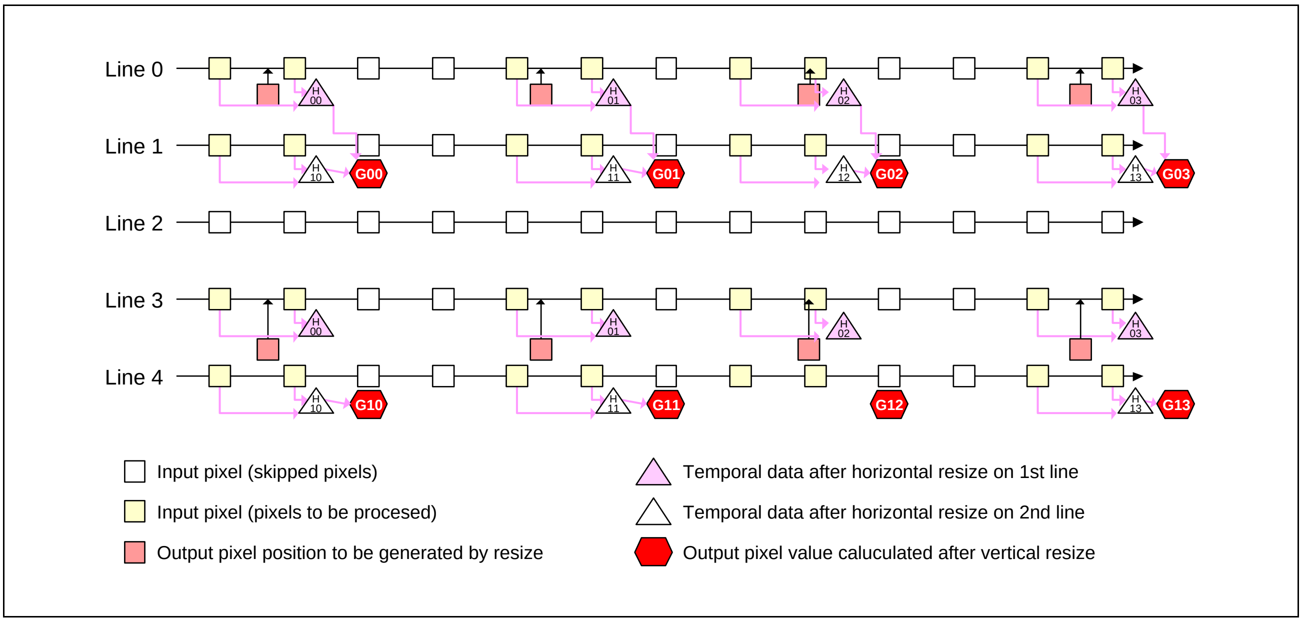
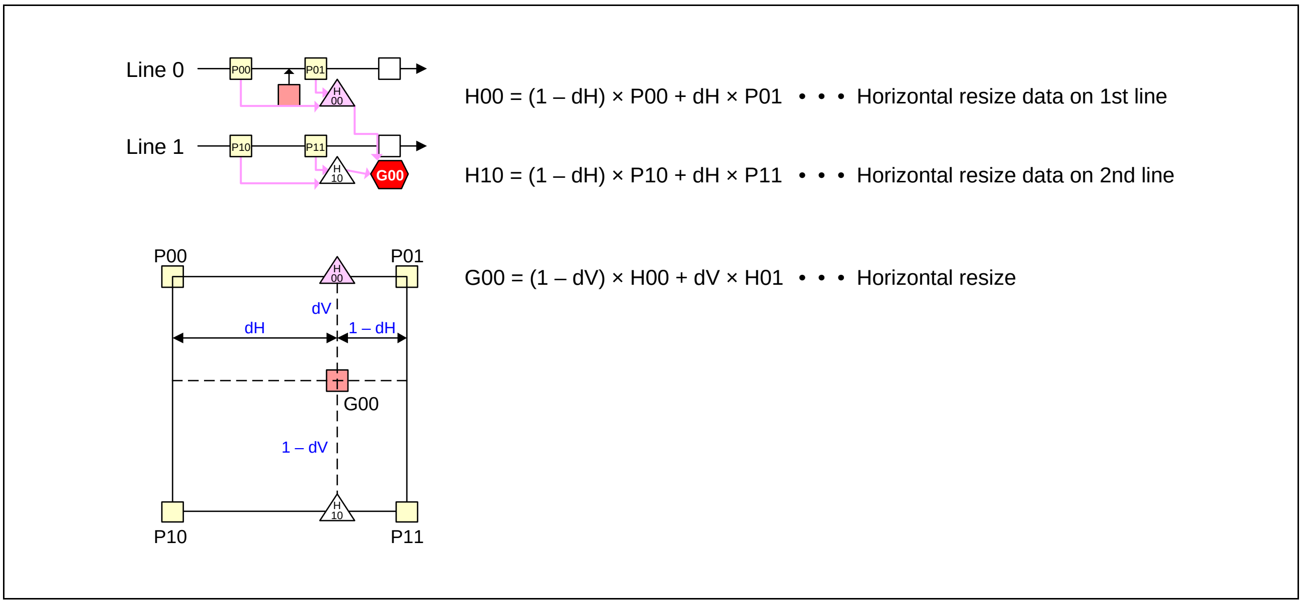


Figure 36.29 Points Bi-Linear Resize

The outline of the calculation procedure of 4-point linear interpolation is shown below.

Although dH and dV are not exist actually, it is used to help understanding for this algorithm. The pixel value of the target pixel position G00 will be calculated from the nearest 4 input pixels, and dH and dV indicate sub pixel position of G00 calculated from Horizontal/Vertical scaling ratio.



$H00 = (1 - dH) \times P00 + dH \times P01$ • • • Horizontal resize data on 1st line

$H10 = (1 - dH) \times P10 + dH \times P11$ • • • Horizontal resize data on 2nd line

$G00 = (1 - dV) \times H00 + dV \times H01$ • • • Horizontal resize

Figure 36.30 Calculation for 4 Points Bi-linear Resize

36.1.5 Write Pixel Formatting (WPF)

WPF converts the resized image data from RS and outputs it to AXI-Master.

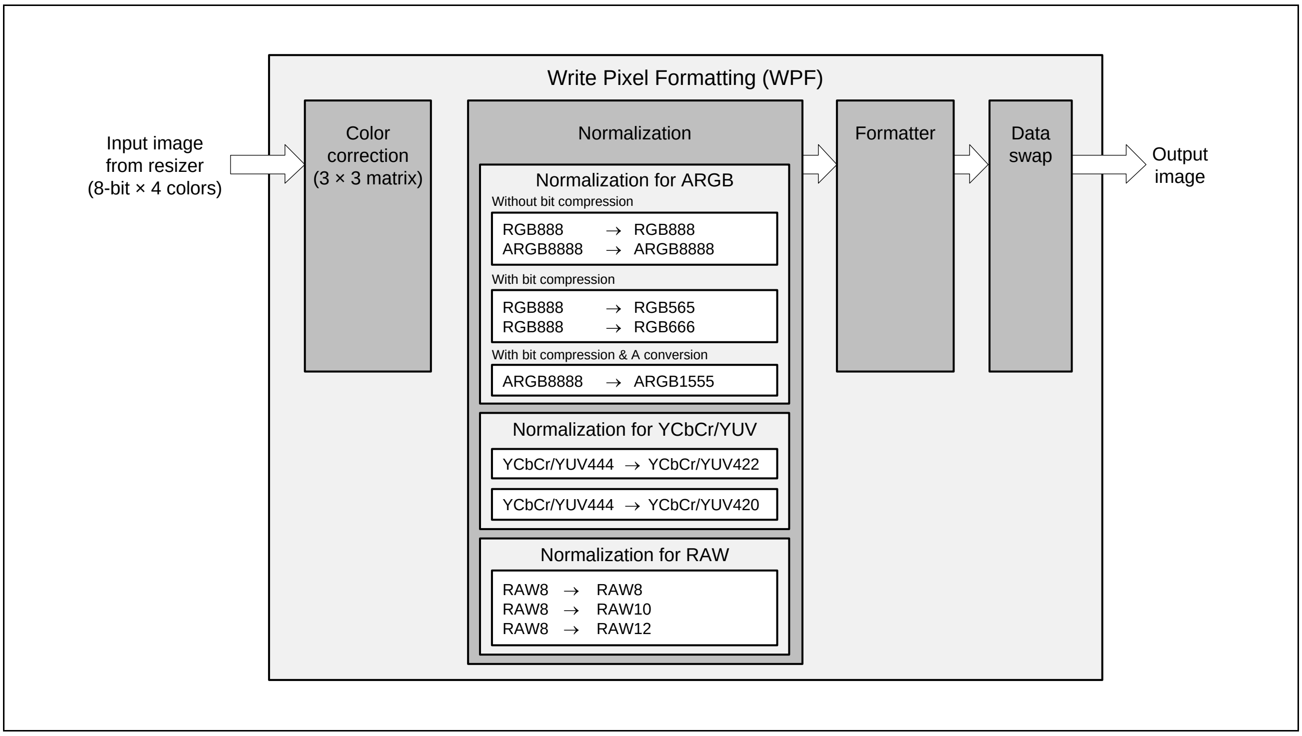


Figure 36.31 WPF Block Diagram

36.1.5.1 Color Correction

This function is color space conversion between RGB and YCbCr and YUV by using 3 × 3 color matrix. Matrix coefficients can be set by registers. The matrix calculation formula and circuit block diagram are shown below.

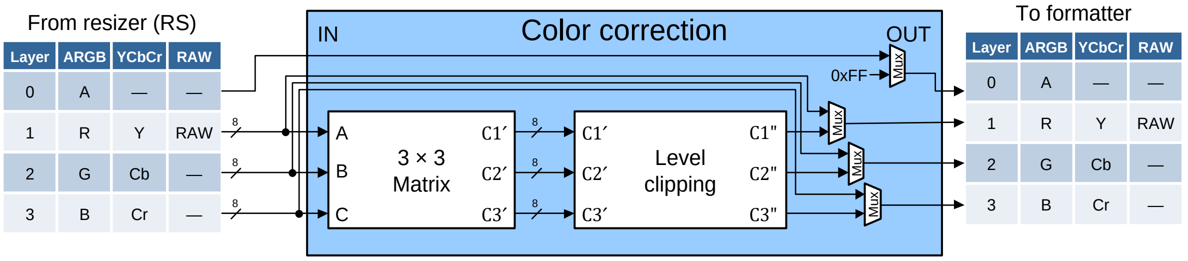
3 × 3 Matrix

$$\begin{bmatrix} C1' \\ C2' \\ C3' \end{bmatrix} = \begin{bmatrix} K11 & K12 & K13 \\ K21 & K22 & K23 \\ K31 & K32 & K33 \end{bmatrix} \cdot \begin{bmatrix} A - OFST\_A1 \\ B - OFST\_B1 \\ C - OFST\_C1 \end{bmatrix} + \begin{bmatrix} OFST\_A2 \\ OFST\_B2 \\ OFST\_C2 \end{bmatrix}$$

Level clipping

$$C1'' = MAX\,(CLPMIN\_A,\ MIN\,(CLPMAX\_A,\ C1'))$$
$$C2'' = MAX\,(CLPMIN\_B,\ MIN\,(CLPMAX\_B,\ C2'))$$
$$C3'' = MAX\,(CLPMIN\_C,\ MIN\,(CLPMAX\_C,\ C3'))$$

Figure 36.32 Color Correction Equation



Layer	ARGB	YCbCr	RAW
0	A	—	—
1	R	Y	RAW
2	G	Cb	—
3	B	Cr	—

Layer	ARGB	YCbCr	RAW
0	A	—	—
1	R	Y	RAW
2	G	Cb	—
3	B	Cr	—

Figure 36.33 Color Correction Block Diagram

Sample setting of color matrix are shown below.

Conversion between RGB and YUV (PAL, SECAM)

$$\begin{bmatrix} Y_{OUT} \\ U_{OUT} \\ V_{OUT} \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R_{in} \\ G_{in} \\ B_{in} \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \qquad \begin{bmatrix} R_{OUT} \\ G_{OUT} \\ B_{OUT} \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 1.140 \\ 1.000 & -0.395 & -0.580 \\ 1.000 & 2.032 & -0.000 \end{bmatrix} \begin{bmatrix} Y_{in} \\ (U_{in} - 128) \\ (V_{in} - 128) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Conversion between RGB and YCbCr ITU-R BT.601 / BT.709 (1250/50/2:1)

$$\begin{bmatrix} Y_{OUT} \\ Cb_{OUT} \\ Cr_{OUT} \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R_{in} \\ G_{in} \\ B_{in} \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \qquad \begin{bmatrix} R_{OUT} \\ G_{OUT} \\ B_{OUT} \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 1.402 \\ 1.000 & -0.344 & -0.714 \\ 1.000 & 1.772 & -0.000 \end{bmatrix} \begin{bmatrix} Y_{in} \\ (Cb_{in} - 128) \\ (Cr_{in} - 128) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Conversion between RGB and YCbCr ITU-R BT.709 (1125/60/2:1)

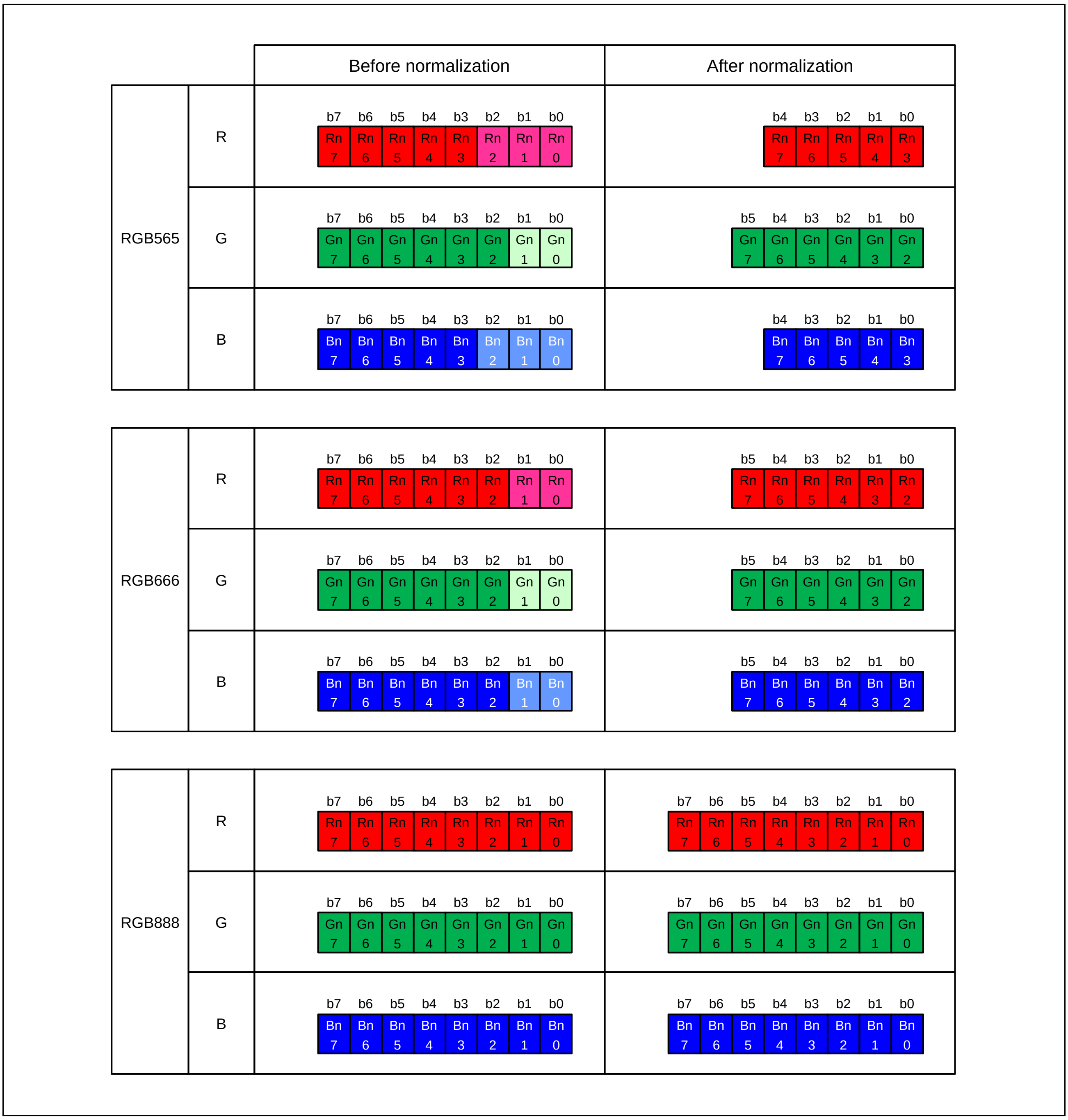
$$\begin{bmatrix} Y_{OUT} \\ Cb_{OUT} \\ Cr_{OUT} \end{bmatrix} = \begin{bmatrix} 0.213 & 0.715 & 0.072 \\ -0.115 & -0.385 & 0.500 \\ 0.500 & -0.454 & -0.046 \end{bmatrix} \begin{bmatrix} R_{in} \\ G_{in} \\ B_{in} \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \qquad \begin{bmatrix} R_{OUT} \\ G_{OUT} \\ B_{OUT} \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 1.575 \\ 1.000 & -0.187 & -0.468 \\ 1.000 & 1.856 & -0.000 \end{bmatrix} \begin{bmatrix} Y_{in} \\ (Cb_{in} - 128) \\ (Cr_{in} - 128) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Conversion from RAW to RGB or YCbCr/YUV (Grayscale)

$$\begin{bmatrix} R_{OUT} \\ G_{OUT} \\ B_{OUT} \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 0.000 \\ 1.000 & 0.000 & 0.000 \\ 1.000 & 0.000 & 0.000 \end{bmatrix} \begin{bmatrix} RAW_{in} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} Y_{OUT} \\ Cb_{OUT} \\ Cr_{OUT} \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 \end{bmatrix} \begin{bmatrix} RAW_{in} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}$$

36.1.5.2 Normalization

This function converts each color data into appropriate bit depth.



		Before normalization	After normalization
RGB565	R	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0	b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3
	G	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0	b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2
	B	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0	b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3
RGB666	R	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0	b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2
	G	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0	b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2
	B	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0	b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2
RGB888	R	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0	b7 b6 b5 b4 b3 b2 b1 b0 Rn7 Rn6 Rn5 Rn4 Rn3 Rn2 Rn1 Rn0
	G	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0	b7 b6 b5 b4 b3 b2 b1 b0 Gn7 Gn6 Gn5 Gn4 Gn3 Gn2 Gn1 Gn0
	B	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0	b7 b6 b5 b4 b3 b2 b1 b0 Bn7 Bn6 Bn5 Bn4 Bn3 Bn2 Bn1 Bn0

Figure 36.34 Normalization for RGB Format

■ ARGB

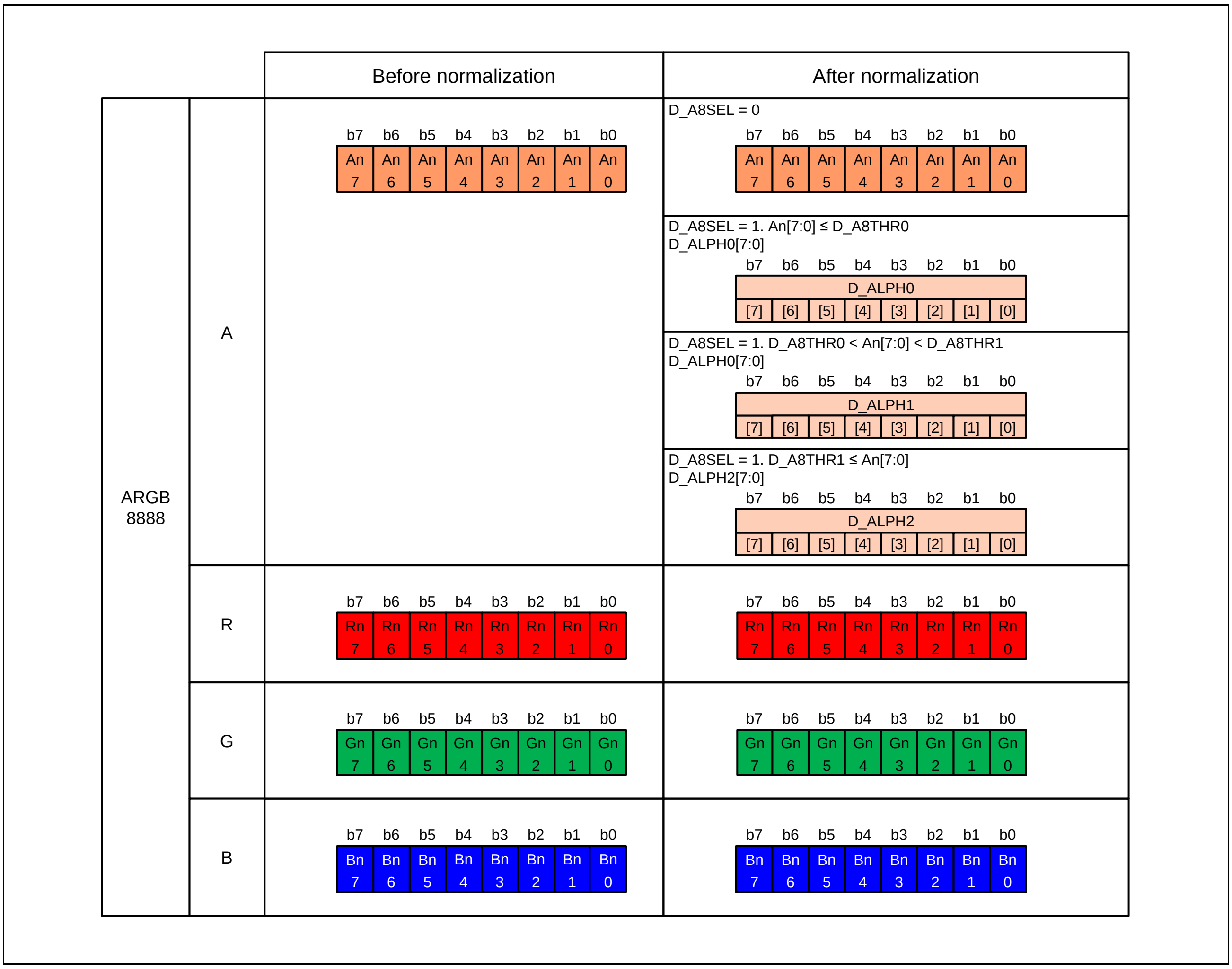


Figure 36.35 Normalization for ARGB8888 Format

“A(An)” of ARGB8888 is normalized according to the following logic.

If ISU_WPF_ALPH_SEL2.D_A8SEL = 0, An is output as it is.

If ISU_WPF_ALPH_SEL2.D_A8SEL = 1, replace it with the following value according to the relationship between ISU_WPF_ALPH_SEL2.D_A8THR0 to 1 and An.

If An ≤ D_A8THR0, replace it with ISU_WPF_ALPH_VAL.D_ALPH0.

If D_A8THR0 < An <D_A8THR1, replace it with ISU_WPF_ALPH_VAL.D_ALPH1.

If D_A8THR1 ≤ An, replace it with ISU_WPF_ALPH_VAL.D_ALPH2.

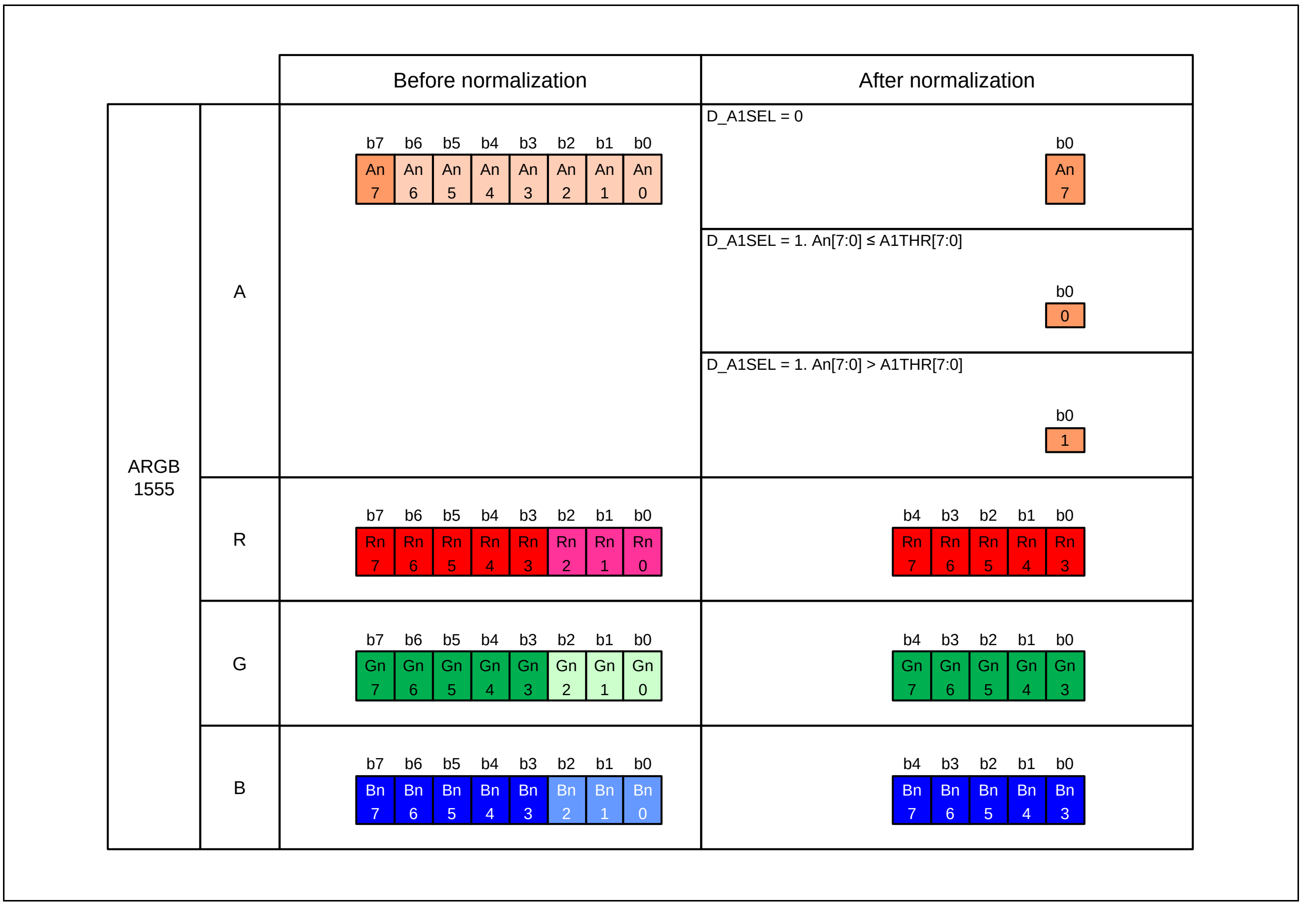


Figure 36.36 Normalization for ARGB1555 Format

“A(An)” of ARGB1555 A is compressed according to the following.

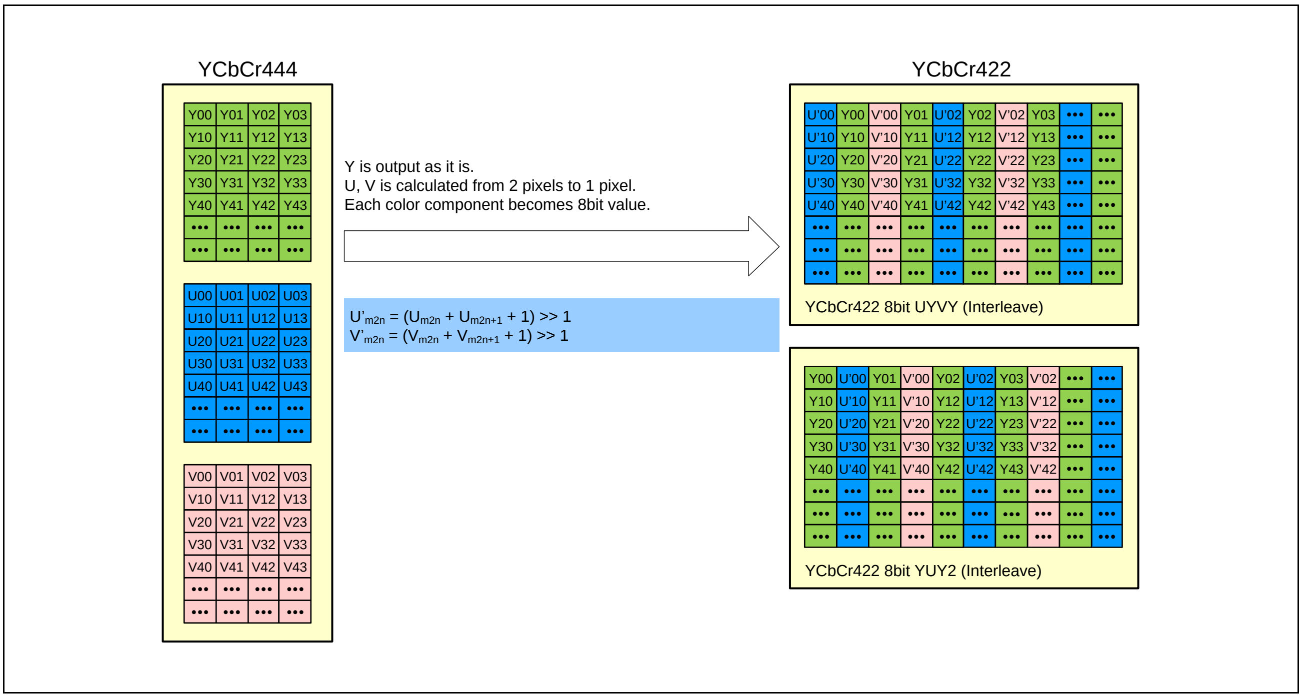
If ISU_WPF_ALPH_SEL1.D_A1SEL = 0, An7 is output.

If ISU_WPF_ALPH_SEL1.D_A1SEL = 1, replace it with the following value according to the relationship between ISU_WPF_ALPH_SEL1.D_A1THR and An.

If An ≤ D_A1THR, 0 is output.

If D_A1THR < An, 1 is output.

■ YCbCr/YUV



Y is output as it is.
U, V is calculated from 2 pixels to 1 pixel.
Each color component becomes 8bit value.

$U'_{m2n} = (U_{m2n} + U_{m2n+1} + 1) >> 1$
$V'_{m2n} = (V_{m2n} + V_{m2n+1} + 1) >> 1$

YCbCr444

Y00	Y01	Y02	Y03
Y10	Y11	Y12	Y13
Y20	Y21	Y22	Y23
Y30	Y31	Y32	Y33
Y40	Y41	Y42	Y43
•••	•••	•••	•••
•••	•••	•••	•••

U00	U01	U02	U03
U10	U11	U12	U13
U20	U21	U22	U23
U30	U31	U32	U33
U40	U41	U42	U43
•••	•••	•••	•••
•••	•••	•••	•••

V00	V01	V02	V03
V10	V11	V12	V13
V20	V21	V22	V23
V30	V31	V32	V33
V40	V41	V42	V43
•••	•••	•••	•••
•••	•••	•••	•••

YCbCr422

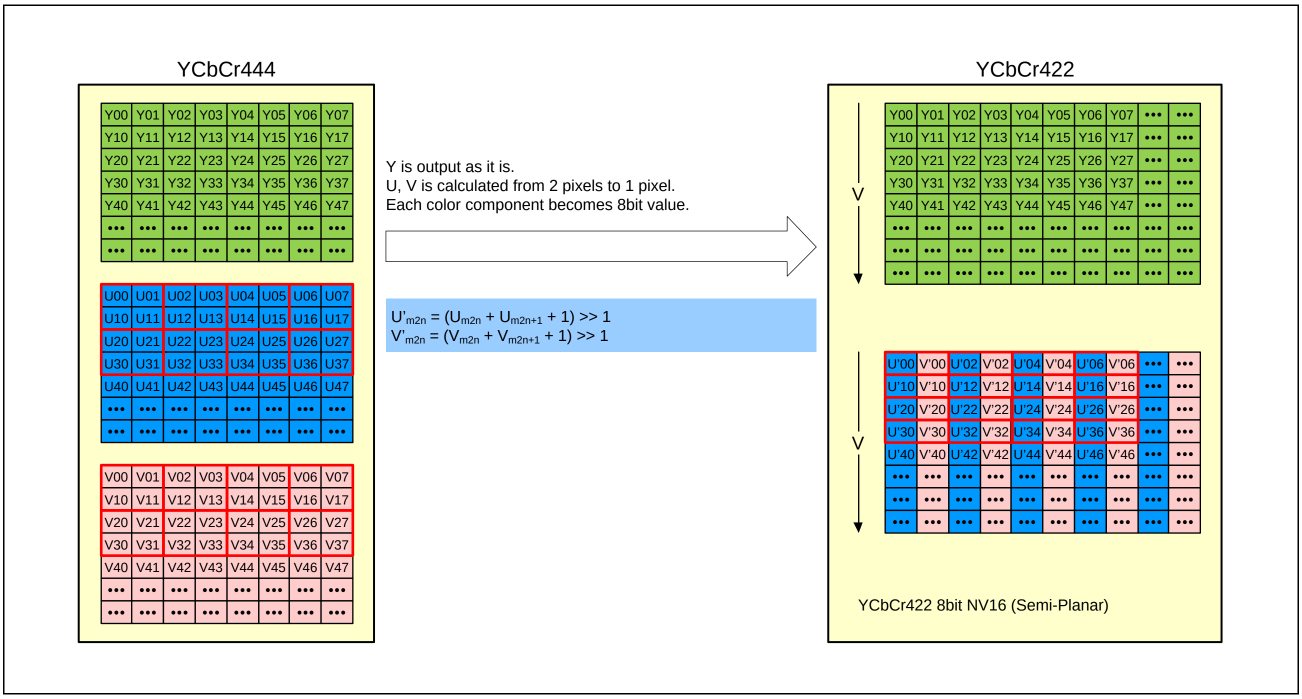
U'00	Y00	V'00	Y01	U'02	Y02	V'02	Y03	•••	•••
U'10	Y10	V'10	Y11	U'12	Y12	V'12	Y13	•••	•••
U'20	Y20	V'20	Y21	U'22	Y22	V'22	Y23	•••	•••
U'30	Y30	V'30	Y31	U'32	Y32	V'32	Y33	•••	•••
U'40	Y40	V'40	Y41	U'42	Y42	V'42	Y43	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••

YCbCr422 8bit UYVY (Interleave)

Y00	U'00	Y01	V'00	Y02	U'02	Y03	V'02	•••	•••
Y10	U'10	Y11	V'10	Y12	U'12	Y13	V'12	•••	•••
Y20	U'20	Y21	V'20	Y22	U'22	Y23	V'22	•••	•••
Y30	U'30	Y31	V'30	Y32	U'32	Y33	V'32	•••	•••
Y40	U'40	Y41	V'40	Y42	U'42	Y43	V'42	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••

YCbCr422 8bit YUY2 (Interleave)

Figure 36.37 Normalization for YCbCr/YUV422 (Interleave) Format



Y is output as it is.
U, V is calculated from 2 pixels to 1 pixel.
Each color component becomes 8bit value.

$U'_{m2n} = (U_{m2n} + U_{m2n+1} + 1) >> 1$
$V'_{m2n} = (V_{m2n} + V_{m2n+1} + 1) >> 1$

YCbCr444

Y00	Y01	Y02	Y03	Y04	Y05	Y06	Y07
Y10	Y11	Y12	Y13	Y14	Y15	Y16	Y17
Y20	Y21	Y22	Y23	Y24	Y25	Y26	Y27
Y30	Y31	Y32	Y33	Y34	Y35	Y36	Y37
Y40	Y41	Y42	Y43	Y44	Y45	Y46	Y47
•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••

U00	U01	U02	U03	U04	U05	U06	U07
U10	U11	U12	U13	U14	U15	U16	U17
U20	U21	U22	U23	U24	U25	U26	U27
U30	U31	U32	U33	U34	U35	U36	U37
U40	U41	U42	U43	U44	U45	U46	U47
•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••

V00	V01	V02	V03	V04	V05	V06	V07
V10	V11	V12	V13	V14	V15	V16	V17
V20	V21	V22	V23	V24	V25	V26	V27
V30	V31	V32	V33	V34	V35	V36	V37
V40	V41	V42	V43	V44	V45	V46	V47
•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••

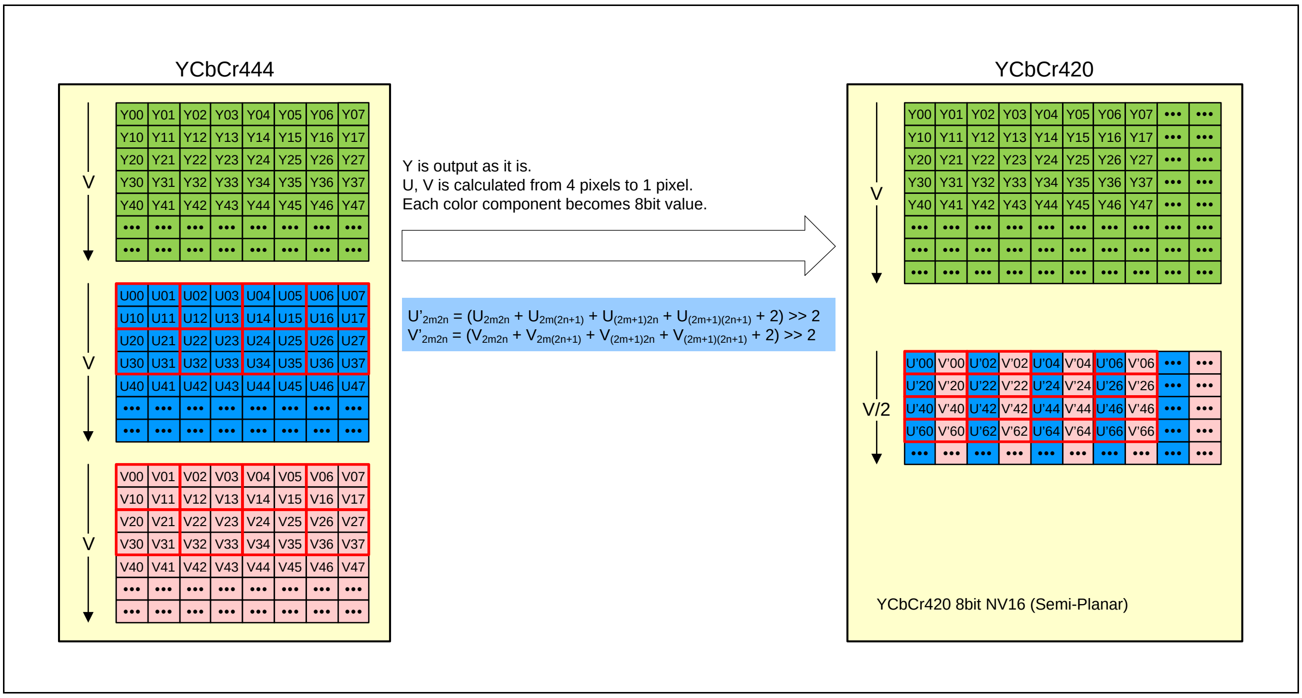
YCbCr422

Y00	Y01	Y02	Y03	Y04	Y05	Y06	Y07	•••	•••
Y10	Y11	Y12	Y13	Y14	Y15	Y16	Y17	•••	•••
Y20	Y21	Y22	Y23	Y24	Y25	Y26	Y27	•••	•••
Y30	Y31	Y32	Y33	Y34	Y35	Y36	Y37	•••	•••
Y40	Y41	Y42	Y43	Y44	Y45	Y46	Y47	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••

U'00	V'00	U'02	V'02	U'04	V'04	U'06	V'06	•••	•••
U'10	V'10	U'12	V'12	U'14	V'14	U'16	V'16	•••	•••
U'20	V'20	U'22	V'22	U'24	V'24	U'26	V'26	•••	•••
U'30	V'30	U'32	V'32	U'34	V'34	U'36	V'36	•••	•••
U'40	V'40	U'42	V'42	U'44	V'44	U'46	V'46	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••
•••	•••	•••	•••	•••	•••	•••	•••	•••	•••

YCbCr422 8bit NV16 (Semi-Planar)

Figure 36.38 Normalization for YCbCr/YUV422 (Semi-Planar) Format

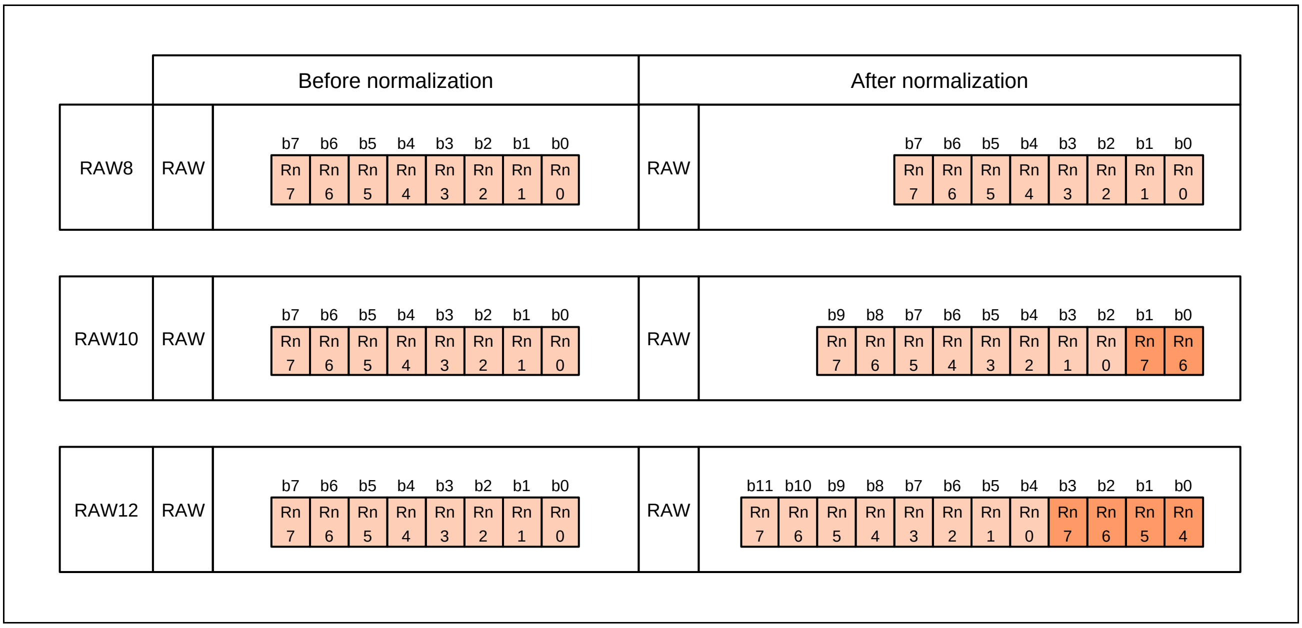


Y is output as it is.
U, V is calculated from 4 pixels to 1 pixel.
Each color component becomes 8bit value.

$U'_{2m2n} = (U_{2m2n} + U_{2m(2n+1)} + U_{(2m+1)2n} + U_{(2m+1)(2n+1)} + 2) >> 2$
$V'_{2m2n} = (V_{2m2n} + V_{2m(2n+1)} + V_{(2m+1)2n} + V_{(2m+1)(2n+1)} + 2) >> 2$

Figure 36.39 Normalization for YCbCr/YUV420 Format

■ RAW



	Before normalization									After normalization												
		b7	b6	b5	b4	b3	b2	b1	b0		b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
RAW8	RAW	Rn 7	Rn 6	Rn 5	Rn 4	Rn 3	Rn 2	Rn 1	Rn 0	RAW					Rn 7	Rn 6	Rn 5	Rn 4	Rn 3	Rn 2	Rn 1	Rn 0
RAW10	RAW	Rn 7	Rn 6	Rn 5	Rn 4	Rn 3	Rn 2	Rn 1	Rn 0	RAW			Rn 7	Rn 6	Rn 5	Rn 4	Rn 3	Rn 2	Rn 1	Rn 0	Rn 7	Rn 6
RAW12	RAW	Rn 7	Rn 6	Rn 5	Rn 4	Rn 3	Rn 2	Rn 1	Rn 0	RAW	Rn 7	Rn 6	Rn 5	Rn 4	Rn 3	Rn 2	Rn 1	Rn 0	Rn 7	Rn 6	Rn 5	Rn 4

Figure 36.40 Normalization for RAW Format

36.1.5.3 Formatter

Following data format are described as 2-/3-/4-/8-byte per unit. Eventually, these data unit are packed in 32-byte (256-bit) each so that it can be handled in next process. Image format setting is indicated by ISU_WPF_FMT.WDFMT[5:0].

■ RGB/ARGB

This function support 8 ARGB/RGB formats as bellow. All formats are “Number of planes = 1”.

Table 36.9 ARGB Support Format

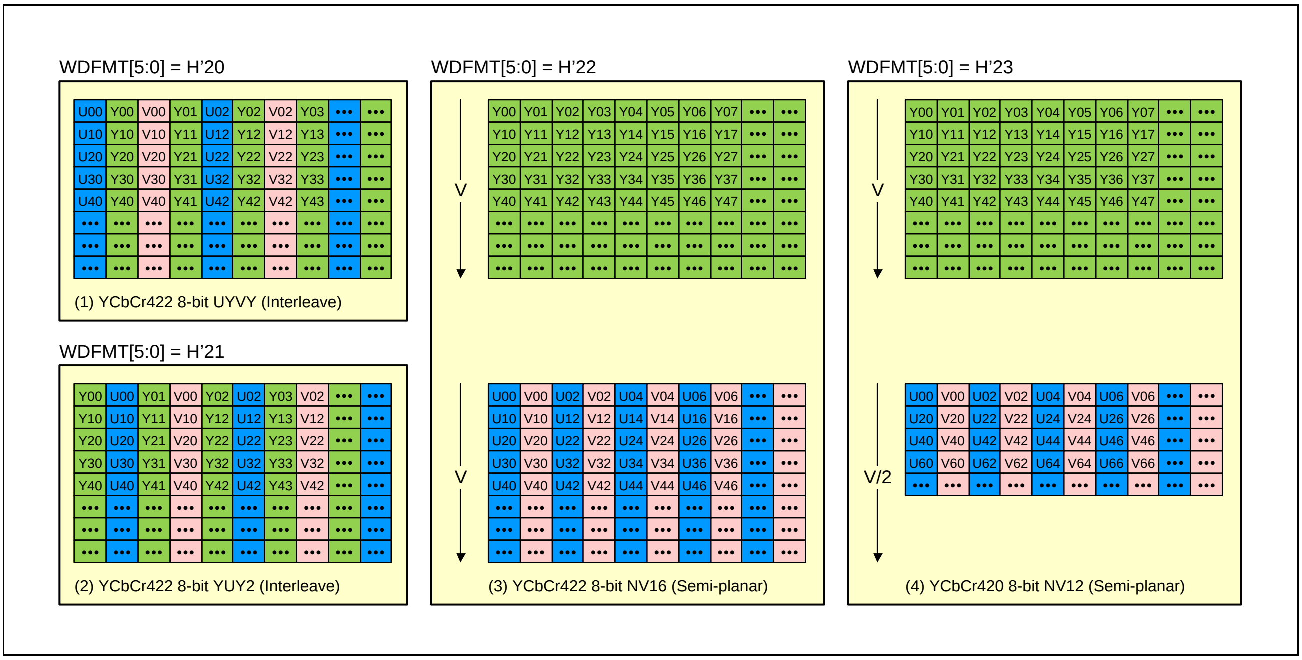
	Format				n								n + 1								n + 2								n + 3							
WDFMT [5:0]	Color Format	Padding	bit/pixel	phase	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
H’01	RGB565	—	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H’05	ARGB8888	—	24	—	A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
H’06	RGBA8888	—		—	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
H’03	RGB888	—		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
				1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
				2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
H’04	BGR888	—	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
				1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
				2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
H’00	ARGB1555	—	15	—	A	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
H’02	BGR666	BGR upper	18	0	▨	▨	B0	B0	B0	B0	B0	B0	▨	▨	G0	G0	G0	G0	G0	G0	▨	▨	R0	R0	R0	R0	R0	R0	▨	▨	B1	B1	B1	B1	B1	B1
				1	▨	▨	G1	G1	G1	G1	G1	G1	▨	▨	R1	R1	R1	R1	R1	R1	▨	▨	B2	B2	B2	B2	B2	B2	▨	▨	G2	G2	G2	G2	G2	G2
				2	▨	▨	R2	R2	R2	R2	R2	R2	▨	▨	B3	B3	B3	B3	B3	B3	▨	▨	G3	G3	G3	G3	G3	G3	▨	▨	R3	R3	R3	R3	R3	R3
H’07	ABGR8888	—	24	—	A	A	A	A	A	A	A	A	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0

■ YCbCr/YUV

This function support 4 YCrCb/YUV formats as bellow.

Table 36.10 YCbCr/YUV Support Format

Data Format	RDFMT[5:0]	Color Format	Alias	Figure Index
Interleave (Number of Planes = 1)	H’20	YCbCr422 8-bit	UYVY	
	H’21	YCbCr422 8-bit	YUY2	
Semi-Planar (Number of Planes = 2)	H’22	YCbCr422 8-bit	NV16	
	H’23	YCbCr420 8-bit	NV12	



(1) YCbCr422 8-bit UYVY (Interleave)

(2) YCbCr422 8-bit YUY2 (Interleave)

(3) YCbCr422 8-bit NV16 (Semi-planar)

(4) YCbCr420 8-bit NV12 (Semi-planar)

Figure 36.41 YCbCr/YUV Support Format

■ RAW

This function support 3 RAW formats as bellow. All formats are “Number of planes = 1”.

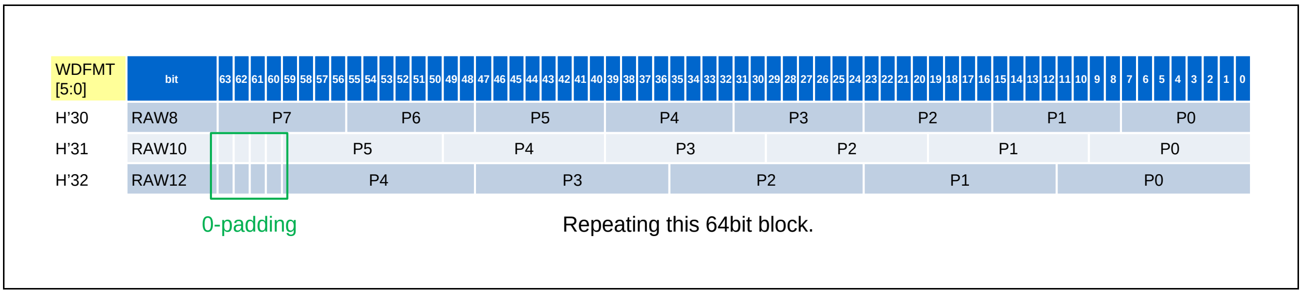
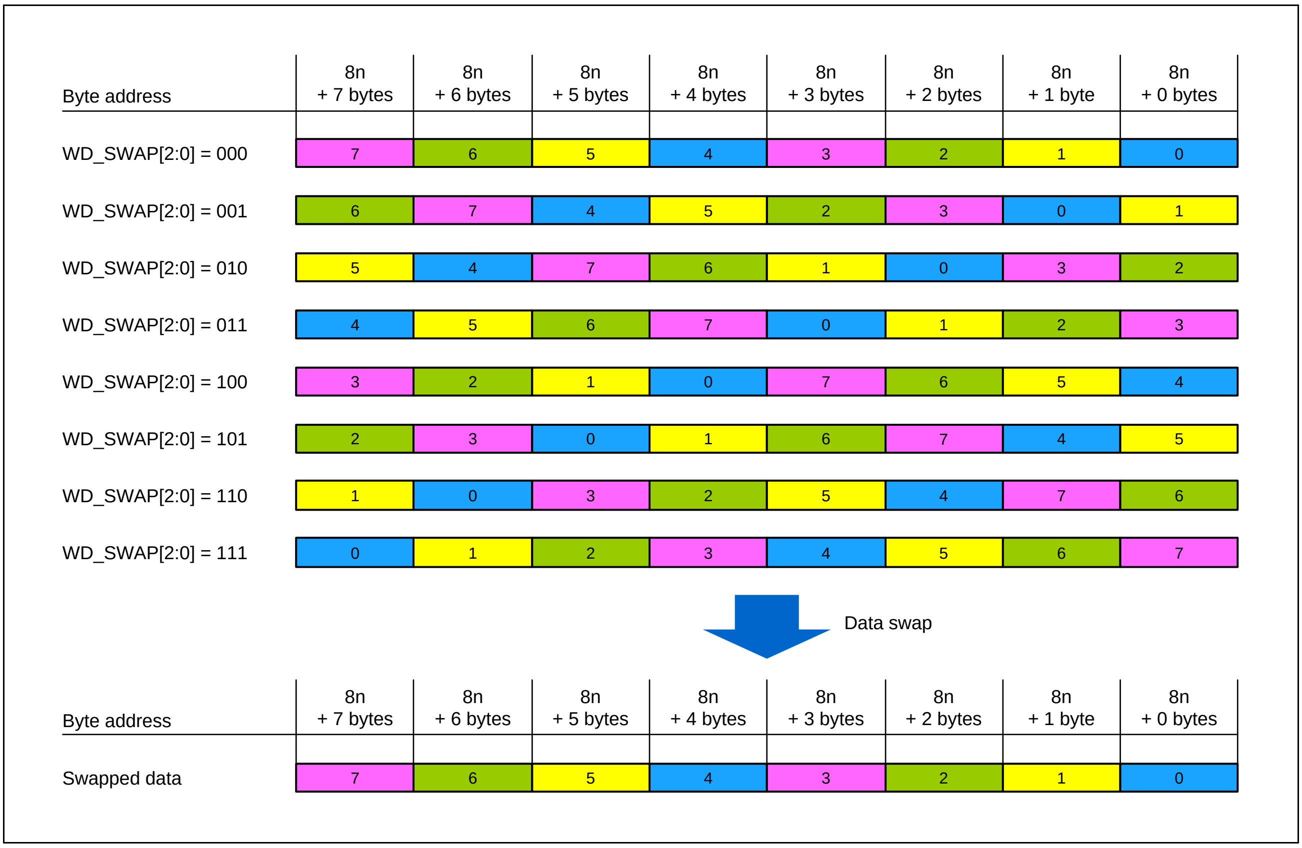


Figure 36.42 RAW Support Format

36.1.5.4 Data Swap (Endian Correction)

This function corrects byte order so that modify endian of data on memory. Each bytes can be arranged any order according the register setting.



Byte address	8n + 7 bytes	8n + 6 bytes	8n + 5 bytes	8n + 4 bytes	8n + 3 bytes	8n + 2 bytes	8n + 1 byte	8n + 0 bytes
WD_SWAP[2:0] = 000	7	6	5	4	3	2	1	0
WD_SWAP[2:0] = 001	6	7	4	5	2	3	0	1
WD_SWAP[2:0] = 010	5	4	7	6	1	0	3	2
WD_SWAP[2:0] = 011	4	5	6	7	0	1	2	3
WD_SWAP[2:0] = 100	3	2	1	0	7	6	5	4
WD_SWAP[2:0] = 101	2	3	0	1	6	7	4	5
WD_SWAP[2:0] = 110	1	0	3	2	5	4	7	6
WD_SWAP[2:0] = 111	0	1	2	3	4	5	6	7

Byte address	8n + 7 bytes	8n + 6 bytes	8n + 5 bytes	8n + 4 bytes	8n + 3 bytes	8n + 2 bytes	8n + 1 byte	8n + 0 bytes
Swapped data	7	6	5	4	3	2	1	0

Figure 36.43 Data Swap Correction

36.2 Register Configuration

The address of the ISU register is represented by the offset address from the base address. ISU base address is as follows:

Base Address: H’0_1084_0000 (Cortex-A55 Address Space)
Base Address: H’4084_0000 (Cortex-M33 Address Space Non-Secure)
Base Address: H’5084_0000 (Cortex-M33 Address Space Secure)

36.2.1 System Management Register

“System Management Register” sets descriptor processing and interrupts control.

The descriptor cannot access these registers.

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
FM Descriptor List Address Registers 0	ISU_FM_DL_STADDH	R/W	H’0000_0000	H’000	32
FM Descriptor List Address Registers 1	ISU_FM_DL_STADDL	R/W	H’0000_0000	H’004	32
FM Frame Control Registers	ISU_FM_FRCON	R/W	H’0000_0000	H’008	32
FM Module Stop Registers	ISU_FM_STOP	R/W	H’0000_0000	H’00C	32
FM Interrupt Enable Registers	ISU_FM_INT_EN	R/W	H’0000_0000	H’010	32
FM Interrupt Status Registers	ISU_FM_INT_STA	R/W	H’0000_0000	H’014	32
AXI Error Action Registers	ISU_AXI_ERACT	R/W	H’0000_0000	H’020	32
AXI FIFO Capability Registers	ISU_AXI_FIFO_CAP	R	H’4000_4000	H’02C	32

36.2.2 Common Register

“Common Register” sets input/output, reduction ratio and color conversion coefficient.

Prohibits write to these registers during frame processing.

The descriptor can access these registers.

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
RPF Source Image Size Registers	ISU_RPF_SRC_SIZE	R/W	H’0000_0000	H’100	32
RPF Source Stride Registers	ISU_RPF_SRC_STRD	R/W	H’0000_0000	H’104	32
RPF Source Plane0 Address Registers 0	ISU_RPF_SRC_ADDH_PL0	R/W	H’0000_0000	H’108	32
RPF Source Plane0 Address Registers 1	ISU_RPF_SRC_ADDL_PL0	R/W	H’0000_0000	H’10C	32
RPF Source Plane1 Address Registers 0	ISU_RPF_SRC_ADDH_PL1	R/W	H’0000_0000	H’110	32
RPF Source Plane1 Address Registers 1	ISU_RPF_SRC_ADDL_PL1	R/W	H’0000_0000	H’114	32
RPF Source Image Format Registers	ISU_RPF_FMT	R/W	H’0000_0000	H’118	32
RPF Source Image UV Format Register	ISU_RPF_UVBIN	R/W	H’0000_0000	H’11C	32
RPF Source Image Data Swap Registers	ISU_RPF_SRC_DSWAP	R/W	H’0000_0000	H’120	32
RPF Source ALPHA Data Selection Registers	ISU_RPF_ALPH_SEL	R/W	H’0000_0000	H’124	32
RPF Source TEST Data Register1	ISU_RPF_SRC_TD1	R/W	H’0000_0000	H’128	32
RPF Source TEST Data Register2	ISU_RPF_SRC_TD2	R/W	H’0000_0000	H’12C	32
RS Scaling Factor Registers 0	ISU_RS_HSCALE	R/W	H’0000_0000	H’140	32
RS Scaling Factor Registers 1	ISU_RS_VSCALE	R/W	H’0000_0000	H’144	32
RS Output Image Start Position Registers	ISU_RS_STPOS	R/W	H’0000_0000	H’148	32
RS Output Image Start Position Tuning Registers	ISU_RS_POS_TUNE	R/W	H’0000_0000	H’14C	32
RS Output Size Crop Registers	ISU_RS_OS_CROP	R/W	H’0000_0000	H’150	32
RS CROP Padding Mode Registers	ISU_RS_PADDMODE	R/W	H’0000_0000	H’154	32
RS CROP Padding Value Registers	ISU_RS_PADDVAL	R/W	H’0000_0000	H’158	32
WPF Destination Plane0 Address Registers 0	ISU_WPF_DST_ADDH_PL0	R/W	H’0000_0000	H’180	32
WPF Destination Plane0 Address Registers 1	ISU_WPF_DST_ADDL_PL0	R/W	H’0000_0000	H’184	32
WPF Destination Plane1 Address Registers 0	ISU_WPF_DST_ADDH_PL1	R/W	H’0000_0000	H’188	32
WPF Destination Plane1 Address Registers 1	ISU_WPF_DST_ADDL_PL1	R/W	H’0000_0000	H’18C	32
WPF Destination Stride Registers	ISU_WPF_DST_STRD	R/W	H’0000_0000	H’190	32
WPF Destination Image Format Registers	ISU_WPF_FMT	R/W	H’0000_0000	H’194	32
WPF Color Correction Control Registers	ISU_WPF_CCOL	R/W	H’0000_0000	H’198	32
WPF Color Correction MUL Coefficient Registers1	ISU_WPF_MUL1	R/W	H’0000_0000	H’19C	32
WPF Color Correction MUL Coefficient Registers2	ISU_WPF_MUL2	R/W	H’0000_0000	H’1A0	32
WPF Color Correction MUL Coefficient Registers3	ISU_WPF_MUL3	R/W	H’0000_0000	H’1A4	32
WPF Color Correction MUL Coefficient Registers4	ISU_WPF_MUL4	R/W	H’0000_0000	H’1A8	32

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
WPF Color Correction MUL Coefficient Registers5	ISU_WPF_MUL5	R/W	H’0000_0000	H’1AC	32
WPF Color Correction MUL Coefficient Registers6	ISU_WPF_MUL6	R/W	H’0000_0000	H’1B0	32
WPF Color Correction Offset Coefficient Registers1	ISU_WPF_OFST1	R/W	H’0000_0000	H’1B4	32
WPF Color Correction Offset Coefficient Registers2	ISU_WPF_OFST2	R/W	H’0000_0000	H’1B8	32
WPF Color Correction Clip Registers1	ISU_WPF_CLP1	R/W	H’0000_0000	H’1BC	32
WPF Color Correction Clip Registers2	ISU_WPF_CLP2	R/W	H’0000_0000	H’1C0	32
WPF Destination Image Data Swap Registers	ISU_WPF_DST_DSWAP	R/W	H’0000_0000	H’1C4	32
WPF Destination ALPHA Selection Registers1	ISU_WPF_ALPH_SEL1	R/W	H’0000_0000	H’1C8	32
WPF Destination ALPHA Selection Registers2	ISU_WPF_ALPH_SEL2	R/W	H’0000_0000	H’1CC	32
WPF Destination ALPHA Value Registers	ISU_WPF_ALPH_VAL	R/W	H’0000_0000	H’1D0	32
AXI Max Burst Length Registers	ISU_AXI_BLEN	R/W	H’000F_000F	H’1F0	32

36.3 Register Descriptions

36.3.1 FM Descriptor List Address Registers 0 (ISU_FM_DL_STADDH)

This register sets the start address [33:32] of the “Descriptor List”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DLADD[33:32]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	DLADD [33:32]	All 0	R/W	Set the start address [33:32] of the “Descriptor-List”.

36.3.2 FM Descriptor List Address Registers 1 (ISU_FM_DL_STADDL)

This register sets the start address [31:0] of the “Descriptor List”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DLADD[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DLADD[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DLADD [31:0]	All 0	R/W	Set the start address [31:0] of the “Descriptor List”. *Note 1.* Set an address that is a multiple of 32-byte.

36.3.3 FM Frame Control Registers (ISU_FM_FRCON)

This register controls the frame processing method selection and processing start.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DESON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	DESON	All 0	R/W	Processing method selection Select “Register Mode” or “Descriptor Mode”. 0: Register Mode When executing in this mode, set the register before starting frame processing. 1: Descriptor Mode When executing in this mode, write a “Descriptor List” before starting frame processing.
15 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	START	All 0	R/W	Frame processing start instruction [Write] 0: NOP 1: Start frame processing [Read] This bit is always read as 0. *Note 1.* If 1 is written during frame processing, it will be ignored. *Note 2.* Also, when 1 is written, the frame end interrupt (ISU_FM_INT_STA.FREND) and descriptor footer read completion interrupt (ISU_FM_INT_STA.DESEND) are automatically cleared.

36.3.4 FM Module Stop Registers (ISU_FM_STOP)

This register controls an emergency stop.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	STOP	All 0	R/W	Emergency stop instruction [Write] 0: NOP 1: Start emergency stop [Read] This bit is always read as 0. *Note 1.* It will be cleared automatically after the emergency stop is completed. If 1 is written during the emergency stop process, it will be ignored. *Note 2.* If you want to stop frame synchronously, you can stop it by setting the footer of the “Descriptor List” not to process the next frame.

36.3.5 FM Interrupt Enable Registers (ISU_FM_INT_EN)

This register controls the interrupt factor.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	AXIRXERRE	—	—	—	AXITXERRE	—	—	—	—	—	—	—	LISTERRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SRSTENDE	—	—	—	—	—	—	DESENDE	FRENDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	AXIRXERRE	All 0	R/W	Control of AXI bus read error interrupt (ISU_INT_ERR) Controls interrupt by “ISU_FM_INT_STA.RRESPERR”. 0: Do not output interrupt factors 1: Output interrupt factor
27 to 25	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	AXITXERRE	All 0	R/W	Control of AXI bus write error interrupt (ISU_INT_ERR) Controls interrupt by “ISU_FM_INT_STA.BRESPERR”. 0: Do not output interrupt factors 1: Output interrupt factor
23 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	LISTERRE	All 0	R/W	Control of “Descriptor List” format violation interrupt (ISU_INT_ERR) Controls interrupt by the number of bytes set in the header exceeding the limit or not being in 8-byte units. 0: Do not output interrupt factors 1: Output interrupt factor
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	SRSTENDE	All 0	R/W	Control of emergency stop completion interrupt (ISU_INT_STOPE) Controls interrupt by the completion of emergency stop. 0: Do not output interrupt factors 1: Output interrupt factor
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DESENDE	All 0	R/W	Control of descriptor footer read completion interrupt (ISU_INT_DESE) Controls interrupt by the completion of descriptor footer read. 0: Do not output interrupt factors 1: Output interrupt factor *Note 1.* When an error occurs due to a format violation of “Descriptor List”, the ISU_INT_DESE interrupt does not occur even if this bit is set to 1.

Bit	Bit Name	Initial Value	R/W	Description
0	FRENDE	All 0	R/W	Control of frame processing end interrupt (ISU_INT_FRE) Controls interrupt by the frame processing end. 0: Not output interrupt factor 1: Output interrupt factor *Note 1.* This bit is updated according to the setting of the footer if the descriptor mode is set.

36.3.6 FM Interrupt Status Registers (ISU_FM_INT_STA)

This register displays the status of the interrupt factors.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RRESPERR[2:0]			—	BRESPERR[2:0]			—	—	—	—	—	—	LISTERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/WC0	R/WC0	R/WC0	R	R/WC0	R/WC0	R/WC0	R	R	R	R	R	R	R/WC0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SRSTEND	—	—	—	—	—	—	DESEND	FREND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/WC0	R	R	R	R	R	R	R/WC0	R/WC0

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 27	RRESPERR [2:0]	All 0	R/WC0	Displays the status of the AXI read response. Each bit means the following factors. [0] EXOKAY 0: EXOKAY no response 1: EXOKAY response [1] SLVERR 0: SLVERR no response 1: SLVERR response [2] DECERR 0: DECERR no response 1: DECERR response *Note 1.* If you want to clear it, please clear 3-bit at the same time.
26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25 to 23	BRESPERR [2:0]	All 0	R	Displays the status of the AXI write response. Each bit means the following factors. [0] EXOKAY 0: EXOKAY no response 1: EXOKAY response [1] SLVERR 0: SLVERR no response 1: SLVERR response [2] DECERR 0: DECERR no response 1: DECERR response *Note 1.* If you want to clear it, please clear 3-bit at the same time.

Bit	Bit Name	Initial Value	R/W	Description
22 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	LISTERR	All 0	R/WC0	Displays the status of “Descriptor List” format violations (ISU_INT_ERR). 0: Descriptor list does not violate format 1: Descriptor list does violate format
15 to 9	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
8	SRSTEND	All 0	R/WC0	Displays the status of the emergency stop completion (ISU_INT_STOPE). 0: Emergency stop not completed 1: Emergency stop completed
7 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	DESEND	All 0	R/WC0	Displays the status of the descriptor footer read completion (ISU_INT_DESE). 0: Descriptor footer read incomplete 1: Descriptor footer read completed
0	FREND	All 0	R/WC0	Displays the status of the frame processing end (ISU_INT_FRE). 0: Frame processing incomplete 1: Frame processing completed

36.3.7 AXI Error Action Registers (ISU_AXI_ERACT)

This register sets the operation of AXI-Master after an AXI error occurs.

For each AXI error condition, choose whether to ignore and continue or stop subsequent transactions.

Transaction continuation settings are used for debugging purposes.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RRESPEXO_ACT	RRESPSLV_ACT	RRESPDEC_ACT	—	BRESPEXO_ACT	BRESPSLV_ACT	BRESPDEC_ACT	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	RRESPEXO_ACT	All 0	R/W	Operation settings after AXI read response EXOKAY occurs 0: Transaction stop 1: Transaction continuation (For debugging)
28	RRESPSLV_ACT	All 0	R/W	Operation settings after AXI read response SLVERR occurs 0: Transaction stop 1: Transaction continuation (For debugging)
27	RRESPDEC_ACT	All 0	R/W	Operation settings after AXI read response DECERR occurs 0: Transaction stop 1: Transaction continuation (For debugging)
26	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	BRESPEXO_ACT	All 0	R/W	Operation settings after AXI write response EXOKAY occurs 0: Transaction stop 1: Transaction continuation (For debugging)
24	BRESPSLV_ACT	All 0	R/W	Operation settings after AXI write response SLVERR occurs 0: Transaction stop 1: Transaction continuation (For debugging)
23	BRESPDEC_ACT	All 0	R/W	Operation settings after AXI write response DECERR occurs 0: Transaction stop 1: Transaction continuation (For debugging)
22 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

36.3.8 AXI FIFO Capability Registers (ISU_AXI_FIFO_CAP)

This register is a register that displays the free space of the FIFO in AXI-Master.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SCAP_PL0[6:0]							—	—	SCAP_PL1[5:0]					
Initial Value	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DCAP_PL0[6:0]							—	—	DCAP_PL1[5:0]					
Initial Value	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 24	SCAP_PL0 [6:0]	H’40	R	Free space in FIFO of the input image “Plane0” 0 to 32 in Semi-Planar mode. 0 to 64 in other modes.
23 to 22	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
21 to 16	SCAP_PL1 [5:0]	All 0	R	Free space in FIFO of the input image “Plane1” 0 to 32 in Semi-Planar mode. 0 in other modes.
15	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
14 to 8	DCAP_PL0 [6:0]	H’40	R	Free space in FIFO of the output image “Plane0” 0 to 32 in Semi-Planar mode. 0 to 64 in other modes.
7 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	DCAP_PL1 [5:0]	All 0	R	Free space in FIFO of the output image “Plane1” 0 to 32 in Semi-Planar mode. 0 in other modes.

36.3.9 RPF Source Image Size Registers (ISU_RPF_SRC_SIZE)

This register sets the size of the input image.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	S_HSIZE[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	S_VSIZE[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	S_HSIZE [11:0]	All 0	R/W	Horizontal size of the input image Set the horizontal size of the input image in pixels. Set the size from 1 to 2800. 0: Setting prohibited 2801 to: Setting prohibited *Note 1.* For YCbCr type, set it to an even number.
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	S_VSIZE [11:0]	All 0	R/W	Vertical size of the input image Set the vertical size of the input image in pixels. Set the size from 1 to 2047. 0: Setting prohibited 2048 to: Setting prohibited *Note 1.* For YCbCr420, set it to an even number.

36.3.10 RPF Source Stride Registers (ISU_RPF_SRC_STRD)

This register sets the stride of the input image.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSTRPL0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSTRPL1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SSTRPL0 [15:0]	All 0	R/W	Set the “Plane0 (Y/RGB/RAW)” stride Set the “Plane0 (Y/RGB/RAW)” stride in bytes. *Note 1.* Set it as a multiple of 32 bytes and at least the number of horizontal pixels (ISU_RPF_SRC_SIZE.S_HSIZE).
15 to 0	SSTRPL1 [15:0]	All 0	R/W	Set the “Plane1 (C(Cb/Cr))” stride Set the “Plane1 (C(Cb/Cr))” stride in bytes. *Note 1.* Set it as a multiple of 32 bytes and at least the number of horizontal pixels (ISU_RPF_SRC_SIZE.S_HSIZE).

36.3.11 RPF Source Plane0 Address Registers 0 (ISU_RPF_SRC_ADDH_PL0)

This register sets the start address [33:32] of the input image of “Plane0”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SADD_PL0 [33:32]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	SADD_PL0 [33:32]	All 0	R/W	Set the start address [33:32] for “Plane0 (Y/RGB/RAW)”.

36.3.12 RPF Source Plane0 Address Registers 1 (ISU_RPF_SRC_ADDL_PL0)

This register sets the start address [31:0] of the input image of “Plane0”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SADD_PL0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SADD_PL0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SADD_PL0 [31:0]	All 0	R/W	Set the start address [31:0] of “Plane0 (Y/RGB/RAW)”. *Note 1.* Set an address that is a multiple of 32-byte.

36.3.13 RPF Source Plane1 Address Registers 0 (ISU_RPF_SRC_ADDH_PL1)

This register sets the start address [33:32] of the input image of “Plane1”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SADD_PL1 [33:32]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R/W	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	SADD_PL1 [33:32]	All 0	R/W	Set the start address [33:32] of “Plane1 (C(Cb/Cr))”.

36.3.14 RPF Source Plane1 Address Registers 1 (ISU_RPF_SRC_ADDL_PL1)

This register sets the start address [31:0] of the input image of “Plane1”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SADD_PL1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SADD_PL1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SADD_PL1 [31:0]	All 0	R/W	Set the start address [31:0] for “Plane1 (C(Cb/Cr))”. *Note 1.* Set an address that is a multiple of 32-byte.

36.3.15 RPF Source Image Format Registers (ISU_RPF_FMT)

This register sets the color format of the input image.

For details, refer to **Section 36.1.3.2, Formatter**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	RDFMT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R/W	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	RDFMT[5:0]	All 0	R/W	Color format of input image The following formats can be configured. H’00: ARGB1555 H’01: RGB565 H’02: BGR666 H’03: RGB888 H’04: BGR888 H’05: ARGB8888 H’06: RGBA8888 H’07: ABGR8888 H’20: YCbCr422 8-bit UYVY H’21: YCbCr422 8-bit YUY2 H’22: YCbCr422 8-bit NV16 H’23: YCbCr420 8-bit NV12 H’30: RAW8 H’31: RAW10 H’32: RAW12 Others: Setting prohibited

36.3.16 RPF Source Image UV Format Registers (ISU_RPF_UVBIN)

This register sets whether to convert CbCr / UV to offset binary.

For details, refer to **Section 36.1.3.2, Formatter**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UVCHG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	UVCHG	All 0	R/W	Offset binary conversion selection for CbCr / UV data 0: No conversion 1: Conversion

36.3.17 RPF Source Image Data Swap Registers (ISU_RPF_SRC_DSWAP)

This register sets the data swap of the input image.

For details, refer to **Section 36.1.3.1, Data Swap (Endian Correction)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RD_SWAP[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	RD_SWAP [2:0]	All 0	R/W	Data swap for input images

36.3.18 RPF Source ALPHA Data Selection Registers (ISU_RPF_ALPH_SEL)

This register sets the extension method from A (Alpha) 1-bit to 8-bit at the time of ARGB1555 input.

For details, refer to **Section 36.1.3.3, Data Extraction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	S_A1SEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S_ALPH1[7:0]								S_ALPH0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	S_A1SEL	All 0	R/W	Setting of how to extend A(Alpha) from 1-bit to 8-bit 0: Copy input Alpha value 1-bit to all 8-bit 1: Select the register value according to the input Alpha value 1 bit S_ALPH0 when Alpha = 0 S_ALPH1 when Alpha = 1
15 to 8	S_ALPH1 [7:0]	All 0	R/W	Extended Alpha value when S_A1SEL = 1 and input Alpha value is 1.
7 to 0	S_ALPH0 [7:0]	All 0	R/W	Extended Alpha value when S_A1SEL = 1 and input Alpha value is 0.

36.3.19 RPF Source TEST Data Registers 1 (ISU_RPF_SRC_TD1)

This register controls the test pattern generation.

For details, refer to **Section 36.1.3.4, Test Pattern Generation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMODE_EN	—	—	—	—	—	—	—	—	—	—	—	GRADA_A	GRADA_R	GRADA_G	GRADA_B
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GRAD_TYPE	GRADA_STEP[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TMODE_EN	All 0	R/W	Control of test pattern generation 0: Normal operation 1: Test pattern generation After one frame of the test pattern is generated, it is automatically cleared to 0. *Note 1.* Setting in descriptor mode is prohibited.
30 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19	GRADA_A	All 0	R/W	A color when generating a test pattern 0: 0 fixed 1: Gradation
18	GRADA_R	All 0	R/W	R color when generating a test pattern 0: 0 fixed 1: Gradation
17	GRADA_G	All 0	R/W	G color when generating a test pattern 0: 0 fixed 1: Gradation
16	GRADA_B	All 0	R/W	B color when generating a test pattern 0: 0 fixed 1: Gradation
15 to 5	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	GRADA_TYPE	All 0	R/W	Gradient type 0: Gradation from 0 to 255 1: Gradation from 255 to 0
3 to 0	GRADA_STEP[3:0]	All 0	R/W	Gradient step *Note 1.* If the setting is 0, the initial value (ISU_RPF_SRC_TD2.INIT_*) is entered.

36.3.20 RPF Source TEST Data Registers 2 (ISU_RPF_SRC_TD2)

This register sets the initial value of each color of ARGB for test pattern generation.

For details, refer to **Section 36.1.3.4, Test Pattern Generation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INIT_A[7:0]								INIT_R[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INIT_G[7:0]								INIT_B[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	INIT_A[7:0]	All 0	R/W	Initial value of A color
23 to 16	INIT_R[7:0]	All 0	R/W	Initial value of R color
15 to 8	INIT_G[7:0]	All 0	R/W	Initial value of G color
7 to 0	INIT_B[7:0]	All 0	R/W	Initial value of B color

36.3.21 RS Scaling Factor Registers 0 (ISU_RS_HSCALE)

This register sets the reduction factor of the input image in the horizontal direction.

For details, refer to **Section 36.1.4.3, Scaling Calculation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	HMANT[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	HFRAC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19 to 16	HMANT[3:0]	All 0	R/W	Horizontal reduction factor (integer part) 0: Setting prohibited
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	HFRAC[11:0]	All 0	R/W	Horizontal multiplication factor (decimal part)

36.3.22 RS Scaling Factor Registers 1 (ISU_RS_VSCALE)

This register sets the reduction factor of the input image in the vertical direction.

For details, refer to **Section 36.1.4.3, Scaling Calculation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	VMANT[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VFRAC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19 to 16	VMANT[3:0]	All 0	R/W	Vertical reduction factor (integer part) 0: Setting prohibited
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	VFRAC[11:0]	All 0	R/W	Vertical reduction factor (decimal part)

36.3.23 RS Output Image Start Position Registers (ISU_RS_STPOS)

This register sets the resize start position.

For details, refer to **Section 36.1.4.1, Resize Start Position**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	HSTART[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VSTART[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26 to 16	HSTART [10:0]	All 0	R/W	Resize start horizontal position Set the resizing start horizontal position by the number of pixels with the upper left of the input image as the origin. Set it with less than the number of input horizontal pixels (ISU_RPF_SRC_SIZE.S_HSIZE).
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10 to 0	VSTART [10:0]	All 0	R/W	Resize start vertical position Set the resizing start vertical position by the number of pixels with the upper left of the input image as the origin. Set it with less than the number of input vertical pixels (ISU_RPF_SRC_SIZE.S_VSIZE).

36.3.24 RS Output Image Start Position Tuning Registers (ISU_RS_POS_TUNE)

This register sets the offset for fine adjustment of the reduction start position.

For details, refer to **Section 36.1.4.2, Sub Pixel Offset**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	HST_TUNE[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VST_TUNE[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
27 to 16	HST_TUNE [11:0]	All 0	R/W	Offset from the resize-starting horizontal position (ISU_RS_STPOS.HSTART) Set the offset to be added to the horizontal position at the start of resizing in 4,096 steps (0 to less than 1 pixel of the input image).
15 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	VST_TUNE [11:0]	All 0	R/W	Offset from the resize-starting vertical position (ISU_RS_STPOS.VSTART) Set the offset to be added to the vertical position at the start of resizing in 4,096 steps (0 to less than 1 pixel of the input image).

36.3.25 RS Output Size Crop Registers (ISU_RS_OS_CROP)

This register sets the crop size of the reduced image.

For details, refer to **Section 36.1.4, Resizer (RS)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	O_HSIZE[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	O_VSIZE[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26 to 16	O_HSIZE [10:0]	All 0	R/W	Horizontal crop size Set the horizontal crop size in pixels. Set it in the range of 1 to 1920. 0: Setting prohibited 1921 to: Setting prohibited *Note 1.* For YCbCr type, set it to an even number.
15 to 11	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10 to 0	O_VSIZE [10:0]	All 0	R/W	Vertical crop size Set the vertical crop size in pixels. Set it in the range of 1 to 1080. 0: Setting prohibited 1081 to: Setting prohibited *Note 1.* For YCbCr420, set it to an even number.

36.3.26 RS CROP Padding Mode Registers (ISU_RS_PADDMODE)

This register sets the padding method when the reduced image is smaller than the set crop size.

For details, refer to **Section 36.1.4, Resizer (RS)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PADDSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PADDSEL	All 0	R/W	Selecting a padding method Set the padding method when the reduced image is smaller than the set crop size. 0: Fill in the padding area by copying the edge pixels after the reduction. 1: Fill in the padding area with the color specified by the register (ISU_RS_PADDVAL.VAL_*).

36.3.27 RS CROP Padding Value Registers (ISU_RS_PADDVAL)

This register sets the color that fills the edges with padding.

For details, refer to **Section 36.1.4, Resizer (RS)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VAL_L3[7:0]								VAL_L2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VAL_L1[7:0]								VAL_L0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	VAL_L3[7:0]	All 0	R/W	The value to fill Layer 3 if the reduced image is smaller than the set crop size.
23 to 16	VAL_L2[7:0]	All 0	R/W	The value to fill Layer 2 if the reduced image is smaller than the set crop size.
15 to 8	VAL_L1[7:0]	All 0	R/W	The value to fill Layer 1 if the reduced image is smaller than the set crop size.
7 to 0	VAL_L0[7:0]	All 0	R/W	The value to fill Layer 0 if the reduced image is smaller than the set crop size.

36.3.28 WPF Destination Plane0 Address Registers 0 (ISU_WPF_DST_ADDH_PL0)

This register sets the start address [33:32] of the output image of “Plane0”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DADD_PL0 [33:32]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	DADD_PL0 [33:32]	All 0	R/W	Set the write start address [33:32] for “Plane0 (Y/RGB/RAW)”.

36.3.29 WPF Destination Plane0 Address Registers 1 (ISU_WPF_DST_ADDL_PL0)

This register sets the start address [33:32] of the output image of “Plane0”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DADD_PL0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DADD_PL0[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DADD_PL0 [31:0]	All 0	R/W	Set the write start address [31:0] for “Plane0 (Y/RGB/RAW)”. *Note 1.* Set an address that is a multiple of 512-byte.

36.3.30 WPF Destination Plane1 Address Registers 0 (ISU_WPF_DST_ADDH_PL1)

This register sets the start address [33:32] of the output image of “Plane1”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DADD_PL1 [33:32]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1 to 0	DADD_PL1 [33:32]	All 0	R/W	Set the write start address [33:32] for “Plane1 (C(Cb/Cr))”.

36.3.31 WPF Destination Plane1 Address Registers 1 (ISU_WPF_DST_ADDL_PL1)

This register sets the start address [31:0] of the output image of “Plane1”.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DADD_PL1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DADD_PL1[31:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DADD_PL1 [31:0]	All 0	R/W	Set the write start address [31:0] for “Plane1 (C(Cb/Cr))”. *Note 1.* Set an address that is a multiple of 512-byte.

36.3.32 WPF Destination Stride Registers (ISU_WPF_DST_STRD)

This register sets the stride of the output image.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTRPL0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTRPL1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	DSTRPL0 [15:0]	All 0	R/W	Set the “Plane0 (Y/RGB/RAW)” stride Set the “Plane0 (Y/RGB/RAW)” stride in bytes. *Note 1.* Set it as a multiple of 32 bytes and at least the number of horizontal crop size (ISU_RS_OS_CROP.O_HSIZE).
15 to 0	DSTRPL1 [15:0]	All 0	R/W	Set the “Plane1 (C(Cb/Cr))” stride Set the “Plane1 (C(Cb/Cr))” stride in bytes. *Note 1.* Set it as a multiple of 32 bytes and at least the number of horizontal crop size (ISU_RS_OS_CROP.O_HSIZE).

36.3.33 WPF Destination Image Format Registers (ISU_WPF_FMT)

This register sets the color format of the output image.

For details, refer to **Section 36.1.5.3, Formatter**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	WDFMT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	WDFMT[5:0]	All 0	R/W	Color format of output image The following formats can be configured. H’00: ARGB1555 H’01: RGB565 H’02: BGR666 H’03: RGB888 H’04: BGR888 H’05: ARGB8888 H’06: RGBA8888 H’07: ABGR8888 H’20: YCbCr422 8-bit UYVY H’21: YCbCr422 8-bit YUY2 H’22: YCbCr422 8-bit NV16 H’23: YCbCr420 8-bit NV12 H’30: RAW8 H’31: RAW10 H’32: RAW12 Others: Setting prohibited

36.3.34 WPF Color Correction Control Registers (ISU_WPF_CCOL)

This register sets the color conversion correction control.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMAT_SEL	CMAT_ASEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CMAT_SEL	0	R/W	Color conversion correction control 0: No conversion 1: Conversion
0	CMAT_ASEL	0	R/W	A (Alpha) color conversion correction control 0: No conversion 1: Conversion

36.3.35 WPF Color Correction MUL Coefficient Registers1 (ISU_WPF_MUL1)

This register sets the matrix value K11 for the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	K11[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	K11[13:0]	All 0	R/W	Color Correction Coefficient K11 Signed fixed point: 4-bit integer part, 10-bit decimal part

36.3.36 WPF Color Correction MUL Coefficient Registers2 (ISU_WPF_MUL2)

This register sets the matrix value K12 and K13 for the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	K12[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	K13[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 16	K12[13:0]	All 0	R/W	Color Correction Coefficient K12 Signed fixed point: 4-bit integer part, 10-bit decimal part
15 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	K13[13:0]	All 0	R/W	Color Correction Coefficient K13 Signed fixed point: 4-bit integer part, 10-bit decimal part

36.3.37 WPF Color Correction MUL Coefficient Registers3 (ISU_WPF_MUL3)

This register sets the matrix value K21 for the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	K21[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	K21[13:0]	All 0	R/W	Color Correction Coefficient K21 Signed fixed point: 4-bit integer part, 10-bit decimal part

36.3.38 WPF Color Correction MUL Coefficient Registers4 (ISU_WPF_MUL4)

This register sets the matrix value K22 and K23 for the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	K22[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	K23[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 16	K22[13:0]	All 0	R/W	Color Correction Coefficient K22 Signed fixed point: 4-bit integer part, 10-bit decimal part
15 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	K23[13:0]	All 0	R/W	Color Correction Coefficient K23 Signed fixed point: 4-bit integer part, 10-bit decimal part

36.3.39 WPF Color Correction MUL Coefficient Registers5 (ISU_WPF_MUL5)

This register sets the matrix value K31 for the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	K31[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	K31[13:0]	All 0	R/W	Color Correction Coefficient K31 Signed fixed point: 4-bit integer part, 10-bit decimal part

36.3.40 WPF Color Correction MUL Coefficient Registers6 (ISU_WPF_MUL6)

This register sets the matrix value K32 and K33 for the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	K32[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	K33[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 16	K32[13:0]	All 0	R/W	Color Correction Coefficient K32 Signed fixed point: 4-bit integer part, 10-bit decimal part
15 to 14	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
13 to 0	K33[13:0]	All 0	R/W	Color Correction Coefficient K33 Signed fixed point: 4-bit integer part, 10-bit decimal part

36.3.41 WPF Color Correction Offset Coefficient Registers1 (ISU_WPF_OFST1)

This register sets the offset 1 of the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFST_A1[7:0]								OFST_B1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFST_C1[7:0]								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OFST_A1 [7:0]	All 0	R/W	Color Correction Offset value A1 Unsigned integer 8-bit
23 to 16	OFST_B1 [7:0]	All 0	R/W	Color Correction Offset value B1 Unsigned integer 8-bit
15 to 8	OFST_C1 [7:0]	All 0	R/W	Color Correction Offset value C1 Unsigned integer 8-bit
7 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

36.3.42 WPF Color Correction Offset Coefficient Registers2 (ISU_WPF_OFST2)

This register sets the offset 2 of the color conversion correction.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFST_A2[7:0]								OFST_B2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFST_C2[7:0]								—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OFST_A2 [7:0]	All 0	R/W	Color Correction Offset value A2 Unsigned integer 8-bit
23 to 16	OFST_B2 [7:0]	All 0	R/W	Color Correction Offset value B2 Unsigned integer 8-bit
15 to 8	OFST_C2 [7:0]	All 0	R/W	Color Correction Offset value C2 Unsigned integer 8-bit
7 to 0	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

36.3.43 WPF Color Correction Clip Registers1 (ISU_WPF_CLP1)

This register sets the maximum and minimum value of the color conversion correction for Layer 1.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLPMAX_A[7:0]								CLPMIN_A[7:0							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
15 to 8	CLPMAX_A [7:0]	All 0	R/W	Color correction maximum clip level value A Set the value to CLPMAX_A ≥ CLPMIN_A
7 to 0	CLPMIN_A [7:0]	All 0	R/W	Color correction minimum clip level value A Set the value to CLPMAX_A ≥ CLPMIN_A

36.3.44 WPF Color Correction Clip Registers2 (ISU_WPF_CLP2)

This register sets the maximum and minimum value of the color conversion correction for Layer 2 and Layer 3.

For details, refer to **Section 36.1.5.1, Color Correction**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CLPMAX_B[7:0]								CLPMIN_B[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLPMAX_C[7:0]								CLPMIN_C[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CLPMAX_B [7:0]	All 0	R/W	Color correction maximum clip level value B Set the value to CLPMAX_B ≥ CLPMIN_B
23 to 16	CLPMIN_B [7:0]	All 0	R/W	Color correction minimum clip level value B Set the value to CLPMAX_B ≥ CLPMIN_B
15 to 8	CLPMAX_C [7:0]	All 0	R/W	Color correction maximum clip level value C Set the value to CLPMAX_C ≥ CLPMIN_C
7 to 0	CLPMIN_C [7:0]	All 0	R/W	Color correction minimum clip level value C Set the value to CLPMAX_C ≥ CLPMIN_C

36.3.45 WPF Destination Image Data Swap Registers (ISU_WPF_DST_DSWAP)

This register sets the data swap of the output image data.

For details, refer to **Section 36.1.5.4, Data Swap (Endian Correction)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	WD_SWAP[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2 to 0	WD_SWAP [2:0]	All 0	R/W	Data swap for output images

36.3.46 WPF Destination ALPHA Selection Registers1 (ISU_WPF_ALPH_SEL1)

This register sets the method of compressing the A (Alpha) value at the ARGB1555 output from 8-bit to 1-bit.

For details, refer to **Section36.1.5.2, Normalization**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	D_A1SEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	D_A1THR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	D_A1SEL	All 0	R/W	Setting of how to compress A (Alpha) from 8-bit to 1-bit 0: Outputs the MSB of alpha value 1: Selects the output according to the alpha value and threshold level Outputs 0b when Alpha ≤ D_A1THR Outputs 1b when Alpha > D_A1THR
15 to 8	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
7 to 0	D_A1THR [7:0]	All 0	R/W	Threshold level value Value used to determine when D_A1SEL = 1.

36.3.47 WPF Destination ALPHA Selection Registers2 (ISU_WPF_ALPH_SEL2)

This register sets the 8-bit A (Alpha) value at the time of ARGB8888 output.

For details, refer to **Section36.1.5.2, Normalization**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	D_A8SEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D_A8THR1[7:0]								D_A8THR0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	D_A8SEL	All 0	R/W	A (Alpha) 8-bit output selection 0 Output Alpha value as it is 1: Outputs the register value according to the alpha value and the threshold level (D_A8THR0 and D_A8THR1) Outputs D_ALPH2 when D_A8THR1≤ Alpha Outputs D_ALPH1 when D_A8THR0 < Alpha < D_A8THR1 Outputs D_ALPH0 when Alpha ≤ D_A8THR0
15 to 8	D_A8THR1 [7:0]	All 0	R/W	Threshold level value 1 Value used to determine which register value is output as Alpha when D_A8SEL = 1. Set the value to D_A8THR0 < D_A8THR1.
7 to 0	D_A8THR0 [7:0]	All 0	R/W	Threshold level value 0 Value used to determine which register value is output as Alpha when D_A8SEL = 1. Set the value to D_A8THR0 < D_A8THR1.

36.3.48 WPF Destination ALPHA Value Registers (ISU_WPF_ALPH_VAL)

This register sets the A (Alpha) value to be output when ARGB8888 and D_A8SEL = 1.

For details, refer to **Section36.1.5.2, Normalization**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	D_ALPH2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D_ALPH1[7:0]								D_ALPH0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	D_ALPH2 [7:0]	All 0	R/W	Conversion Alpha value 2 The value to be output when D_A8THR1 ≤ Alpha.
15 to 8	D_ALPH1 [7:0]	All 0	R/W	Conversion Alpha value 1 The value to be output when D_A8THR0 < Alpha < D_A8THR1.
7 to 0	D_ALPH0 [7:0]	All 0	R/W	Conversion Alpha value 0 The value to be output when Alpha ≤ D_A8THR0.

36.3.49 AXI Max Burst Length Registers (ISU_AXI_BLEN)

This register sets the upper limit to the burst length of the AXI-Master.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	ARLEN_MAX[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	AWLEN_MAX[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
19 to 16	ARLEN_ MAX[3:0]	H'F	R/W	Upper limit of read data burst length H'3: Up to 4-beat burst H'7: Up to 8-beat burst H'F: Up to 16-beat burst Others: Setting prohibited
15 to 4	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3 to 0	AWLEN_ MAX[3:0]	H'F	R/W	Upper limit of write data burst length H'3: Up to 4-beat burst H'7: Up to 8-beat burst H'F: Up to 16-beat burst Others: Setting prohibited

36.4 Operation

36.4.1 Start Processing

After setting various registers, write 1 to the start instruction bit to start.

At that time, start it together with the setting of whether to execute by the descriptor method.

Table 36.11 Explanation for ISU FM Frame Control Registers (ISU_FM_FRCON)

ICU_FM_FRCON		Description
Bit 16	DESON	0: Execute with Descriptor read OFF. 1: Execute with Descriptor read ON.
Bit 0	START	When the frame processing start is executed, the frame completion interrupt and descriptor capture completion interrupt are automatically cleared. [Write] 0: NOP 1: Start Frame Processing [Read] Always Zero is read.

36.4.1.1 Descriptor List Address

In the case of the descriptor method (DESON = 1 and START is set to 1), the access to acquire the descriptor list is performed first. The access destination is set in advance in the following register.

- FM Descriptor List Address Register0 (ISU_FM_DL_STADDH [2: 0])
- FM Descriptor List Address Register1 (ISU_FM_DL_STADDL [31: 0])

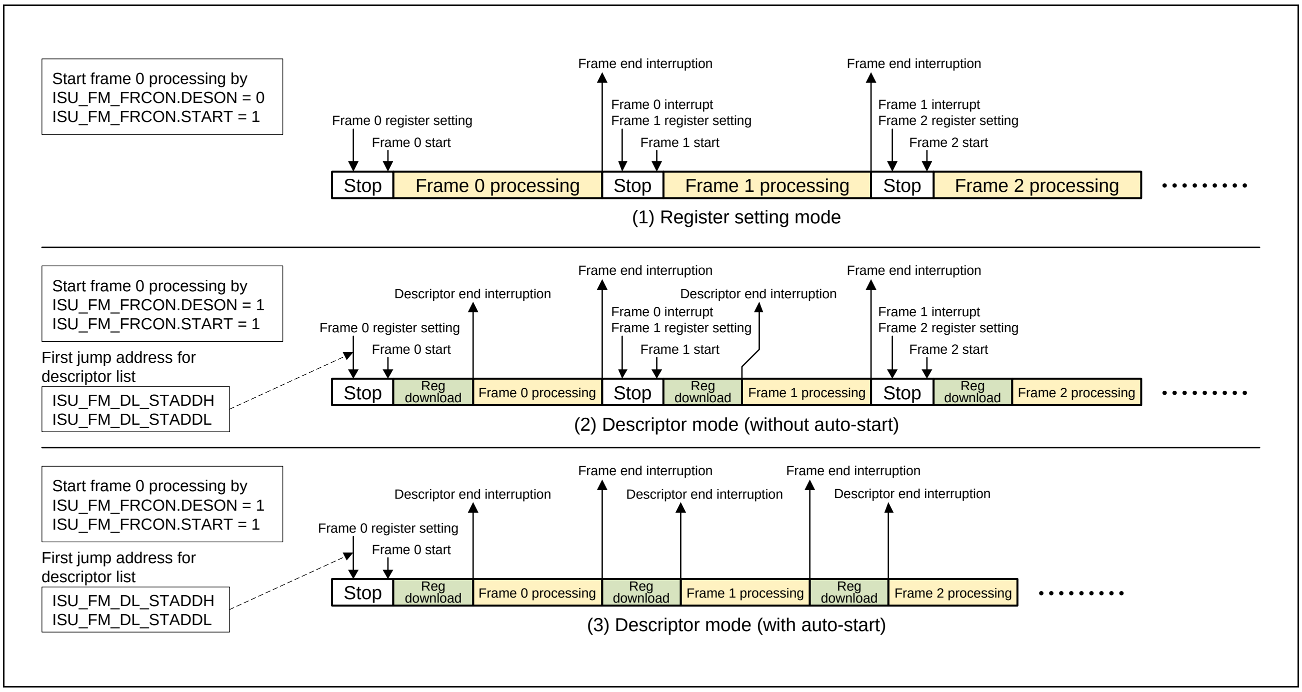


Figure 36.44 Relationship Between Register ISU_FM_FRCON Setting and Frame Processing Mode

36.4.1.2 Frame End Interruption

- When started with ISU_FM_FRCON.DESON = 0
  - An interrupt is generated when one frame processing is completed inside the ISU.
  - Completion of 1-frame processing shall meet the following conditions.
    1) Write the reduced data from the AXI bus and receive the response of the response channel.
    2) The FIFO pointer of the input image is in the initial state (empty state), and the FIFO pointer of the output image is also in the initial state (empty state).
    3) You have not accidentally started capturing the image of the next frame.
  - When the descriptor is not used, only one interrupt is held instead of two interrupts.

- When started with ISU_FM_FRCON.DESON = 1
  - One frame processing by the descriptor is completed, and an interrupt is issued in the descriptor list. An interrupt is generated when it is enabled (see the corresponding bit in the footer).
  - In the descriptor method, it advances to the next frame without doing anything after the interrupt occurs. In this case, the interrupt remains on.
  - If an interrupt is issued in the next frame, it means that two interrupts have occurred. In this case, the interrupt is cleared by clearing the interrupt twice (clearing from the oldest one).
  - ISU retains the latest two interrupt information.

36.4.2 Stop Processing

There are two ways to stop it.

- Normal stop
  - When started with ISU_FM_FRCON.DESON = 0
    It will end automatically when the processing of one frame is completed. After completion, the read and write FIFOs on the AXI-Master side will also be empty.
  - When started with ISU_FM_FRCON.DESON = 1
    If there is no instruction to execute the next frame in the footer of the descriptor, it will automatically end after executing one frame. After completion, the read and write FIFOs on the AXI-Master side will also be empty.

- Emergency stop

Emergency stop will be stopped by register setting from APB.

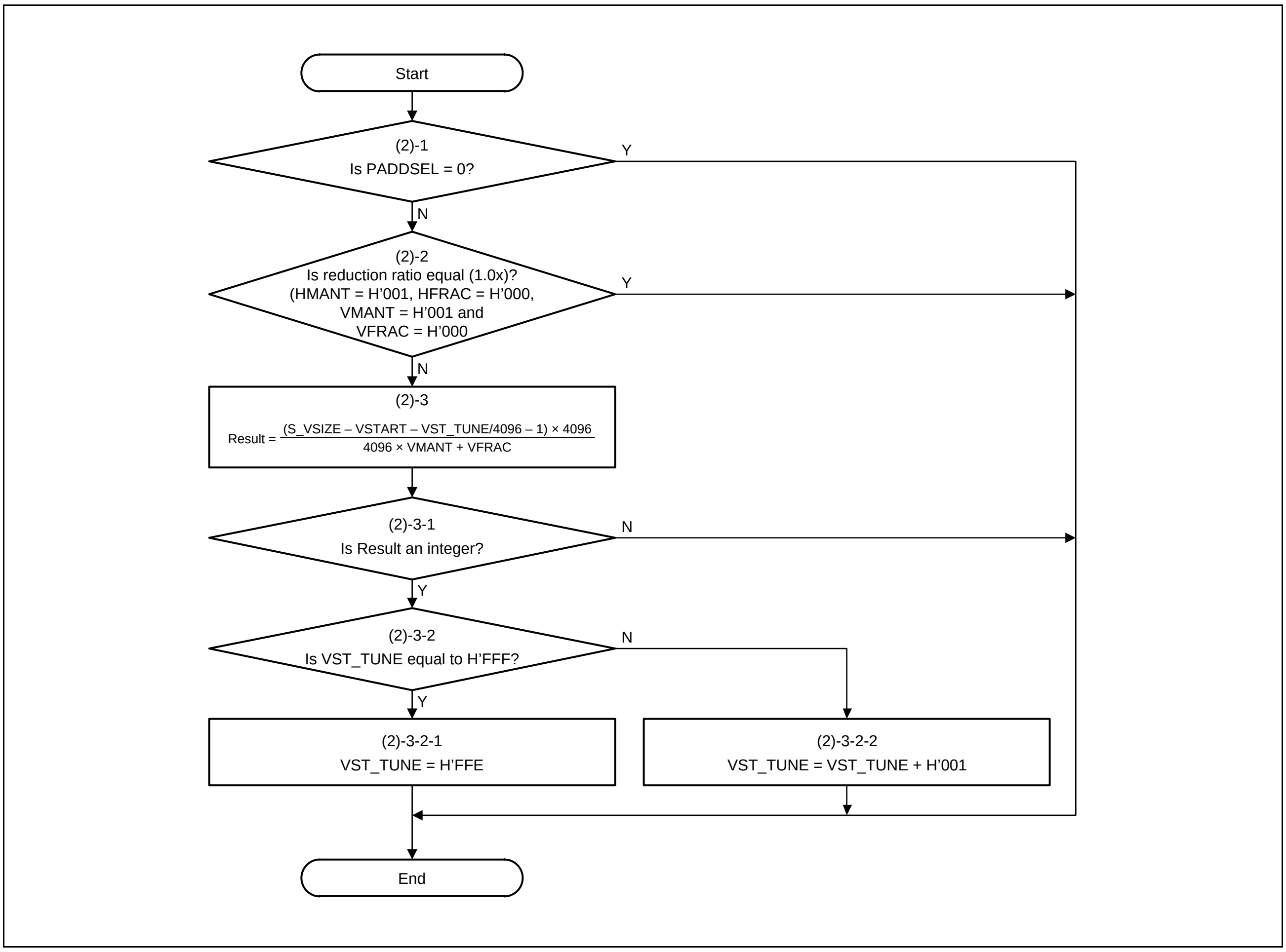
In response to the instruction from APB, the transfer to the bus in process of AXI-Master is executed until a correct response is made, and after it is completed, an interrupt signal is issued. After that, when restarting from the system side, ISU system reset is always applied from the outside and executed from initialization.

If the APB instructs the emergency stop again during the emergency stop process, only the first emergency stop instruction will be valid.

36.4.3 Setting the vertical size, start position and reduction factor

Set the input image size, start position, and reduction ratio before calculating the reduction by Resizer (RS).

Before setting the parameters, please check and adjust the following settings for the vertical direction.



(1) List of related registers

ISU_RPF_SRC_SIZE. S_VSIZE[11:0]
ISU_RS_STPOS.VSTART[10:0]
ISU_RS_POS_TUNE.VST_TUNE[11:0]
ISU_RS_VSCALE.VMANT[3:0]
ISU_RS_VSCALE.VFRAC[11:0]
ISU_RS_HSCALE.HMANT[3:0]
ISU_RS_HSCALE.HFRAC[11:0]
ISU_RS_PADDMODE.PADDSEL

(2) Confirmation and determination of setting values

(2)-1 When PADDSEL = 0
There is no need to change those register parameters.

(2)-2 When the reduction ratio is equal (1.0x)
In other words, when HMANT = H'1, HFRAC = H'000, VMANT = H'1 and VFRAC = H'000
There is no need to change those register parameters.

(2)-3 When the setting value is other than the above (other than equal)
Perform the following calculations.

$$\mathrm{Result} = \frac{(\mathrm{S\_VSIZE} - \mathrm{VSTART} - \mathrm{VST\_TUNE}/4096 - 1) \times 4096}{4096 \times \mathrm{VMANT} + \mathrm{VFRAC}}$$

(2)-3-1 If the calculation result (Result) is not an integer
There is no need to change those register parameters.

(2)-3-2 If the calculation result (Result) is an integer

(2)-3-2-1 If VST_TUNE is equal to H’FFF
VST_TUNE = H’FFE

(2)-3-2-2 If VST_TUNE is not equal to H’FFF
VST_TUNE = VST_TUNE + H’001 (Add 1 to correct the start position)

36.5 Usage Notes

This section shows examples of converting image size and color format using the image scaling unit (ISU).

36.5.1 Interlace to Progressive (IP) Conversion

Although ISU does not have dedicated IP Conversion feature, it is possible to realize similar result by controlling image output address.

Since the ISU does not have functionality to read two images alternately, IP conversion is handled by arranging top and bottom images in a grid pattern by setting the output stride for two lines and shifting the output start address. The output stride must be set to a value in consideration of restrictions for 512-byte boundary of the output start address.

For example, if the output data size for one line is a 300-byte image, 600 bytes are required for two lines. However, when converting to Interface to Progressive, the output stride must be a multiple of 512Byte because the start address of the Bottom line must be set to the 512Byte boundary as well as the start address of the TOP line.

Since this method simply realizes IP Conversion by using the output address control, please note there is not any image processing between Top line and Bottom line.

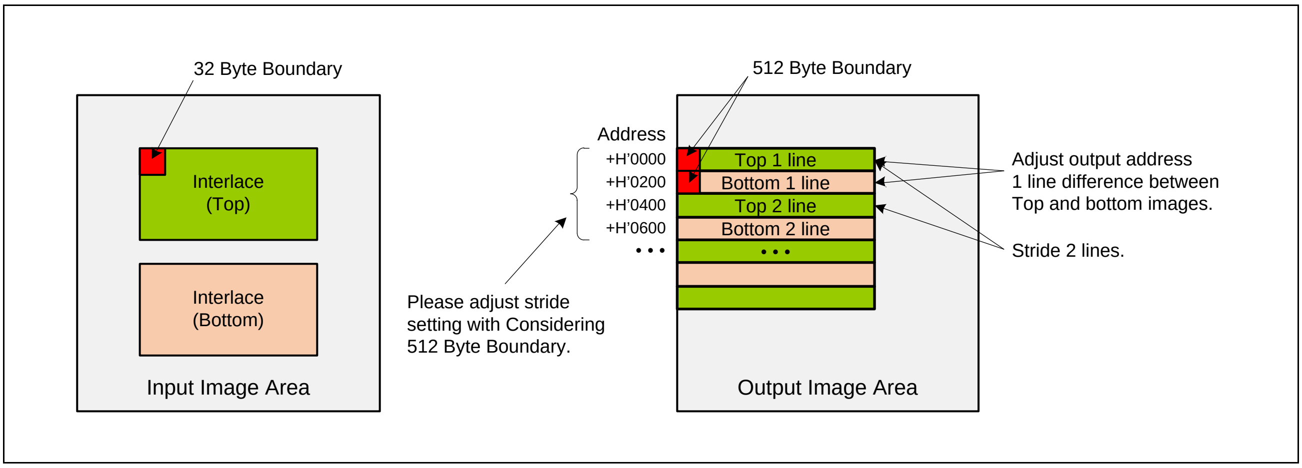


Figure 36.45 Schematic Diagram of IP Conversion

36.5.2 Image Size Conversion

36.5.2.1 NTSC (720 × 480) → WQVGA (400 × 240) with Preserving Aspect Ratio

Set as follows to handle image size conversion.

Table 36.12 Image Size Conversion Example 1

Item	Description
Assuming that the following image is input from the CRU	
Input size	720 × 480
Input format	YCbCr422 8-bit UYVY, Interleave * For IP conversion, see **Section 36.5.1**.
Input start address	H'0_5000_0000 (InterlaceTop) H'0_5010_0000 (InterlaceBottom) H'0_5020_0000 (Progressive)
Input stride	Plane 0 = H'5A0
ISU setting	
Output (CROP) size	380 × 240 (Right-end 20 pixels padding)
Output format	YCbCr422 8-bit UYVY, Interleave
Output start address	Plane 0 = H'0_6000_000
Output stride	Plane 0 = H'300
Processing method	Descriptor method with automatic start of the next frame
Color format conversion	None
Descriptor address	Table 1: H'8000_0000 Table 2: H'8000_0100 Table 3: H'8000_0200

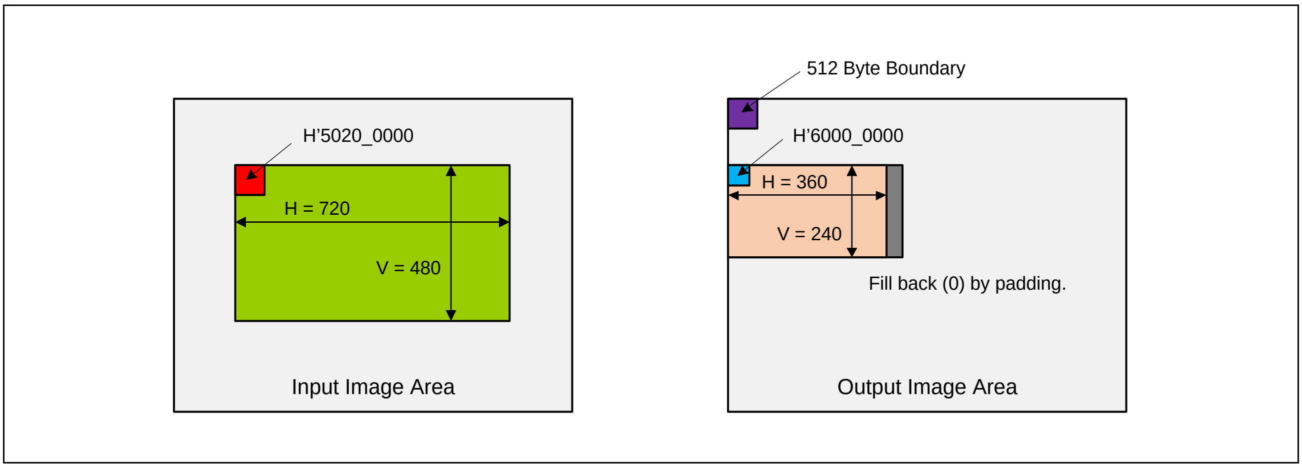


Figure 36.46 Image Size Conversion Example 1

Obtain the multiplication coefficient in the horizontal direction.

720 / 360 = 2.0
HMANT (integer part) = 2 = H’2
HFRAC (decimal part) = 0.0 × 4096 (Q12) = 0.0 = H’0

Obtain the multiplication coefficient in the vertical direction.

480 / 240 = 2.0
VMANT (integer part) = 2 = H’2
VFRAC (decimal part) = 0.0 × 4096 (Q12) = 0.0 = H’0

Note: HFRAC and VFRAC are set by unsigned fixed point Q12 and are truncated after the decimal point.

Set PADDSEL and VAL_L* to make the right-end 20 pixels a black image.

PADDSEL = H’1 (filled with the color specified when the reduced image is smaller than the specified CROP)
VAL_L1, VAL_L2, and VAL_L3 correspond to Y, Cb, and Cr respectively.
Set VAL_L1 = H’10, L2 = H’80, and L3 = H’80 for black images with YCbCr.

Table 36.13 Image Size Conversion Example 1 (Descriptor Table 1, InterlaceTop Image) (1/2)

Concept	Address	Set Value	Comment
Header	H’80000000	H’00000088	Total number of bytes (register address + data) = H’88
Register setting main body	H’80000004	H’00000100	ISU_RPF_SRC_SIZE address
	H’80000008	H’02D000F0	Input size H = 720, V = 240
	H’8000000C	H’00000104	ISU_RPF_SRC_STRD address
	H’80000010	H’05A00000	Plane 0 input stride H’05A0
	H’80000014	H’00000108	ISU_RPF_SRC_ADDH_PL0 address
	H’80000018	H’00000000	Upper 3 bits of Plane 0 input start address H’0
	H’8000001C	H’0000010C	ISU_RPF_SRC_ADDL_PL0 address
	H’80000020	H’50000000	Lower 32 bits of Plane 0 output start address H’5000_0000
	H’80000024	H’00000118	ISU_RPF_FMT address
	H’80000028	H’00000020	Input format YCbCr422 8-bit UYVY (Interleave)
	H’8000002C	H’00000140	ISU_RS_HSCALE address
	H’80000030	H’00010000	Multiplication coefficient in the horizontal direction HMANT = H’1, HFRAC = H’000
	H’80000034	H’00000144	ISU_RS_VSCALE address
	H’80000038	H’00010000	Multiplication coefficient in the vertical direction VMANT = H’1, VFRAC = H’000
	H’8000003C	H’00000150	ISU_RS_OS_CROP address
	H’80000040	H’02D000F0	CROP size H = 720, V = 240
	H’80000044	H’00000180	ISU_WPF_DST_ADDH_PL0 address
	H’80000048	H’00000000	Upper 3 bits of Plane 0 output start address H’0
	H’8000004C	H’00000184	ISU_WPF_DST_ADDL_PL0 address
	H’80000050	H’50200000	Lower 32 bits of Plane 0 output start address H’5020_0000
	H’80000054	H’00000190	ISU_WPF_DST_STRD address
	H’80000058	H’0C000000	Output stride H’0C00
	H’8000005C	H’00000194	ISU_WPF_FMT address
	H’80000060	H’00000020	Output format YCbCr422 8-bit UYVY (Interleave)
	H’80000064	H’0000019C	ISU_WPF_MUL1 address

Table 36.13 Image Size Conversion Example 1 (Descriptor Table 1, InterlaceTop Image) (2/2)

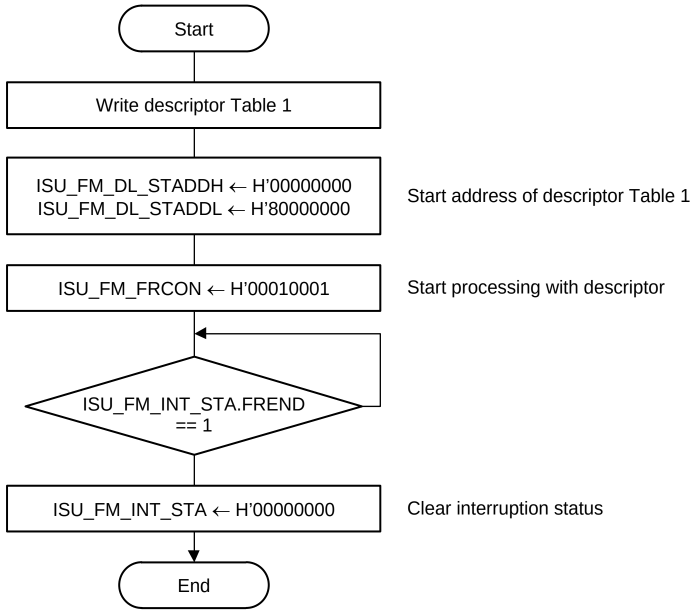
Concept	Address	Set Value	Comment
Register setting main body	H'80000068	H'00000400	K11 = H'400
	H'8000006C	H'000001A8	ISU_WPF_MUL4 address
	H'80000070	H'04000000	K22 = H'400
	H'80000074	H'000001B0	ISU_WPF_MUL6 address
	H'80000078	H'00000400	K33 = H'400
	H'8000007C	H'000001BC	ISU_WPF_CLP1 address
	H'80000080	H'0000FF00	CLP (MAX/MIN)_A = H'FF, H'00
	H'80000084	H'000001C0	ISU_WPF_CLP2 address
	H'80000088	H'FF00FF00	CLP (MAX/MIN)_B = H'FF, H'00, C = H'FF, H'00
Footer	H'8000008C	H'00000000	Since "No automatic start of the next frame" is selected in end of the Footer, the set value is ignored.
	H'80000090	H'00000000	Since "No automatic start of the next frame" is selected in end of the Footer, the set value is ignored.
	H'80000094	H'00000002	No automatic start of the next frame with a frame end interrupt

Table 36.14 Image Size Conversion Example 1 (Descriptor Table 2, InterlaceBottom Image)

Concept	Address	Set Value	Comment
Header	H'80000100	H'00000010	Total number of bytes (register address + data) = H'10
Register setting main body	H'80000104	H'0000010C	ISU_RPF_SRC_ADDL_PL0 address
	H'80000108	H'50100000	Plane 0 input start address H'0_5010_0000
	H'8000010C	H'00000184	ISU_WPF_DST_ADDL_PL0 address
	H'80000110	H'50200600	Plane 0 output start address H'0_5020_0600
Footer	H'80000114	H'00000000	Since "No automatic start of the next frame" is selected in end of the Footer, the set value is ignored.
	H'80000118	H'00000000	Since "No automatic start of the next frame" is selected in end of the Footer, the set value is ignored.
	H'8000011C	H'00000002	No automatic start of the next frame with a frame end interrupt

Table 36.15 Image Size Conversion Example 1 (Descriptor Table 3, Progressive Image)

Concept	Address	Set Value	Comment
Header	H'80000200	H'00000088	Total number of bytes (register address + data) = H'50
Register setting main body	H'80000204	H'00000100	ISU_RPF_SRC_SIZE address
	H'80000208	H'02D001E0	Input size H = 720, V = 480
	H'8000020C	H'00000104	ISU_RPF_SRC_STRD address
	H'80000210	H'06000000	Plane 0 input stride H'0600
	H'80000214	H'0000010C	ISU_RPF_SRC_ADDL_PL0 address
	H'80000218	H'50200000	Plane 0 input start address H'0_5020_0000
	H'8000021C	H'00000140	ISU_RS_HSCALE address
	H'80000220	H'00020000	Multiplication coefficient in the horizontal direction HMANT = H'2, HFRAC = H'000
	H'80000224	H'00000144	ISU_RS_VSCALE address
	H'80000228	H'00020000	Multiplication coefficient in the vertical direction VMANT = H'2, VFRAC = H'000
	H'8000022C	H'00000150	ISU_RS_OS_CROP address
	H'80000230	H'017C00F0	CROP size H = 380, V = 240
	H'80000234	H'00000154	ISU_RS_PADDMODE address
	H'80000238	H'00000001	Padding area is filled with the specified PADDSEL = 1 register value.
	H'8000023C	H'00000158	ISU_RS_PADDVAL address
	H'80000240	H'80801000	VAL_L3 = H'80, L2 = H'80, L1 = H'10, L0 = H'00
	H'80000244	H'00000184	ISU_WPF_DST_ADDL_PL0 address
	H'80000248	H'60000000	Plane 0 output start address H'0_6000_0000
	H'8000024C	H'00000190	ISU_WPF_DST_STRD address
	H'80000250	H'03000000	Output stride H'0300
Footer	H'80000254	H'00000000	Since "No automatic start of the next frame" is selected in end of the Footer, the set value is ignored.
	H'80000258	H'00000000	Since "No automatic start of the next frame" is selected in end of the Footer, the set value is ignored.
	H'8000025C	H'00000002	No automatic start of the next frame with a frame end interrupt



Note: Repeat same operations for descriptor Table 2 & 3.

Figure 36.47 Flowchart of Image Size Conversion Example 1

36.5.2.2 NTSC (720 × 480) → WQVGA (400 × 240) without Preserving Aspect Ratio

Set as follows to handle image size conversion.

Table 36.16 Image Size Conversion Example 2

Item	Description
Assuming that the following image is input from the CRU	
Input size	720 × 480
Input format	YCbCr422 8-bit UYVY, Interleave * For IP conversion, see **Section 36.5.1**.
Input start address	H'0_5000_0000 (InterlaceTop) H'0_5010_0000 (InterlaceBottom) H'0_5020_0000 (Progressive)
Input stride	Plane 0 = H'5A0
ISU setting	
Output (CROP) size	400 × 240
Output format	YCbCr422 8-bit UYVY, Interleave
Output start address	Plane 0 = H'0_6000_000
Output stride	Plane 0 = H'320
Processing method	Descriptor method without automatic start of the next frame
Color format conversion	None
Descriptor address	Table 1: H'8000_0000 Table 2: H'8000_0100 Table 3: H'8000_0200

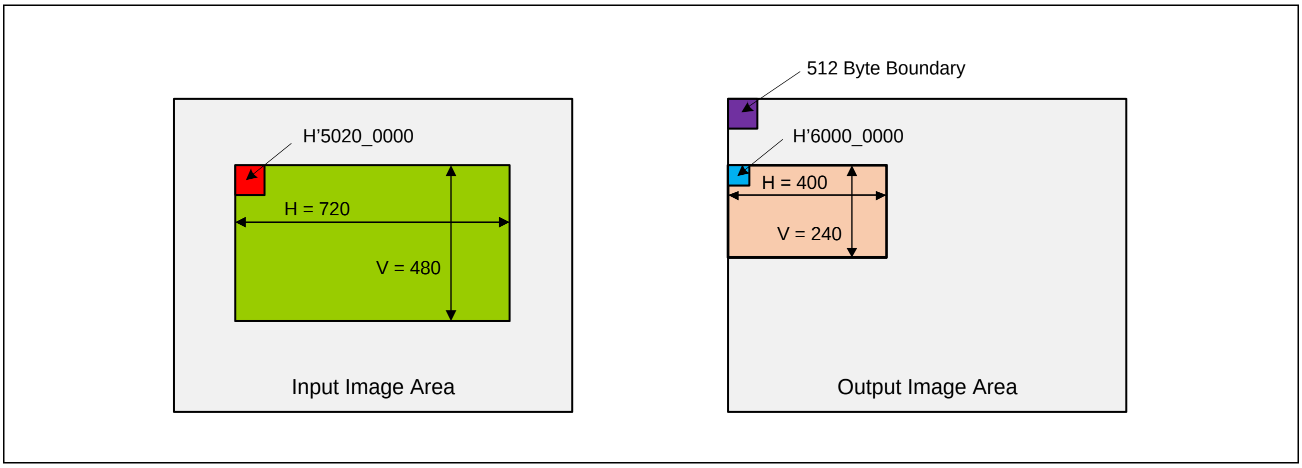


Figure 36.48 Image Size Conversion Example 2

Obtain the multiplication coefficient in the horizontal direction.

720 / 400 = 1.8
HMANT (integer part) = 1 = H’1
HFRAC (decimal part) = 0.8 × 4096 (Q12) = 3276.8 = H’CCC

Obtain the multiplication coefficient in the vertical direction.

480 / 240 = 2.0
VMANT (integer part) = 2 = H’2
VFRAC (decimal part) = 0.0 × 4096 (Q12) = 0.0 = H’0

Note: HFRAC and VFRAC are set by unsigned fixed point Q12 and are truncated after the decimal point.

Table 36.17 Image Size Conversion Example 2 (Descriptor Table 1, InterlaceTop Image) (1/2)

Concept	Address	Set Value	Comment
Header	H’80000000	H’00000088	Total number of bytes (register address + data) = H’88
Register setting main body	H’80000004	H’00000100	ISU_RPF_SRC_SIZE address
	H’80000008	H’02D000F0	Input size H = 720, V = 240
	H’8000000C	H’00000104	ISU_RPF_SRC_STRD address
	H’80000010	H’05A00000	Plane 0 input stride H’05A0
	H’80000014	H’00000108	ISU_RPF_SRC_ADDH_PL0 address
	H’80000018	H’00000000	Upper 3 bits of Plane 0 input start address H’0
	H’8000001C	H’0000010C	ISU_RPF_SRC_ADDL_PL0 address
	H’80000020	H’50000000	Lower 32 bits of Plane 0 output start address H’5000_0000
	H’80000024	H’00000118	ISU_RPF_FMT address
	H’80000028	H’00000020	Input format YCbCr422 8-bit UYVY (Interleave)
	H’8000002C	H’00000140	ISU_RS_HSCALE address
	H’80000030	H’00010000	Multiplication coefficient in the horizontal direction HMANT = H’1, HFRAC = H’000
	H’80000034	H’00000144	ISU_RS_VSCALE address
	H’80000038	H’00010000	Multiplication coefficient in the vertical direction VMANT = H’1, VFRAC = H’000
	H’8000003C	H’00000150	ISU_RS_OS_CROP address
	H’80000040	H’02D000F0	CROP size H = 720, V = 240
	H’80000044	H’00000180	ISU_WPF_DST_ADDH_PL0 address
	H’80000048	H’00000000	Upper 3 bits of Plane 0 output start address H’0
	H’8000004C	H’00000184	ISU_WPF_DST_ADDL_PL0 address
	H’80000050	H’50200000	Lower 32 bits of Plane 0 output start address H’5020_0000
	H’80000054	H’00000190	ISU_WPF_DST_STRD address
	H’80000058	H’0C000000	Output stride H’0C00
	H’8000005C	H’00000194	ISU_WPF_FMT address
	H’80000060	H’00000020	Output format YCbCr422 8-bit UYVY (Interleave)
	H’80000064	H’0000019C	ISU_WPF_MUL1 address
	H’80000068	H’00000400	K11 = H’400
	H’8000006C	H’000001A8	ISU_WPF_MUL4 address
	H’80000070	H’04000000	K22 = H’400
	H’80000074	H’000001B0	ISU_WPF_MUL6 address
	H’80000078	H’00000400	K33 = H’400
	H’8000007C	H’000001BC	ISU_WPF_CLP1 address

Table 36.17 Image Size Conversion Example 2 (Descriptor Table 1, InterlaceTop Image) (2/2)

Concept	Address	Set Value	Comment
	H’80000080	H’0000FF00	CLP (MAX/MIN)_A = H’FF, H’00
	H’80000084	H’000001C0	ISU_WPF_CLP2 address
	H’80000088	H’FF00FF00	CLP (MAX/MIN)_B = H’FF, H’00, C = H’FF, H’00
Footer	H’8000008C	H’80000100	Lower 32 bits of the start address in the next descriptor table H’8000_0100
	H’80000090	H’00000000	Upper 3 bits of the start address in the next descriptor table H’0
	H’80000094	H’00000001	Automatic start of the next frame without a frame end interrupt

Table 36.18 Image Size Conversion Example 2 (Descriptor Table 2, InterlaceBottom Image)

Concept	Address	Set Value	Comment
Header	H’80000100	H’00000010	Total number of bytes (register address + data) = H’10
Register setting main body	H’80000104	H’0000010C	ISU_RPF_SRC_ADDL_PL0 address
	H’80000108	H’50100000	Plane 0 input start address H’0_5010_0000
	H’8000010C	H’00000184	ISU_WPF_DST_ADDL_PL0 address
	H’80000110	H’50200600	Plane 0 output start address H’0_5020_0600
Footer	H’80000114	H’80000200	Lower 32 bits of the start address in the next descriptor table H’8000_0100
	H’80000118	H’00000000	Upper 3 bits of the start address in the next descriptor table H’0
	H’8000011C	H’00000001	Automatic start of the next frame without a frame end interrupt

Table 36.19 Image Size Conversion Example 2 (Descriptor Table 3, Progressive Image)

Concept	Address	Set Value	Comment
Header	H’80000200	H’00000040	Total number of bytes (register address + data) = H’40
Register setting main body	H’80000204	H’00000100	ISU_RPF_SRC_SIZE address
	H’80000208	H’02D001E0	Input size H = 720, V = 480
	H’8000020C	H’00000104	ISU_RPF_SRC_STRD address
	H’80000210	H’06000000	Plane 0 input stride H’0600
	H’80000214	H’0000010C	ISU_RPF_SRC_ADDL_PL0 address
	H’80000218	H’50200000	Plane 0 input start address H’0_5020_0000
	H’8000021C	H’00000140	ISU_RS_HSCALE address
	H’80000220	H’00010CCC	Multiplication coefficient in the horizontal direction HMANT = H’1, HFRAC = H’CCC
	H’80000224	H’00000144	ISU_RS_VSCALE address
	H’80000228	H’00020000	Multiplication coefficient in the vertical direction VMANT = H’2, VFRAC = H’000
	H’8000022C	H’00000150	ISU_RS_OS_CROP address
	H’80000230	H’019000F0	CROP size H = 400, V = 240
	H’80000234	H’00000184	ISU_WPF_DST_ADDL_PL0 address
	H’80000238	H’60000000	Plane 0 output start address H’0_6000_0000
	H’8000023C	H’00000190	ISU_WPF_DST_STRD address
	H’80000240	H’03200000	Output stride H’0320
Footer	H’80000244	H’00000000	Since “No automatic start of the next frame” is selected in end of the Footer, the set value is ignored.
	H’80000248	H’00000000	Since “No automatic start of the next frame” is selected in end of the Footer, the set value is ignored.
	H’8000024C	H’00000002	No automatic start of the next frame with a frame end interrupt

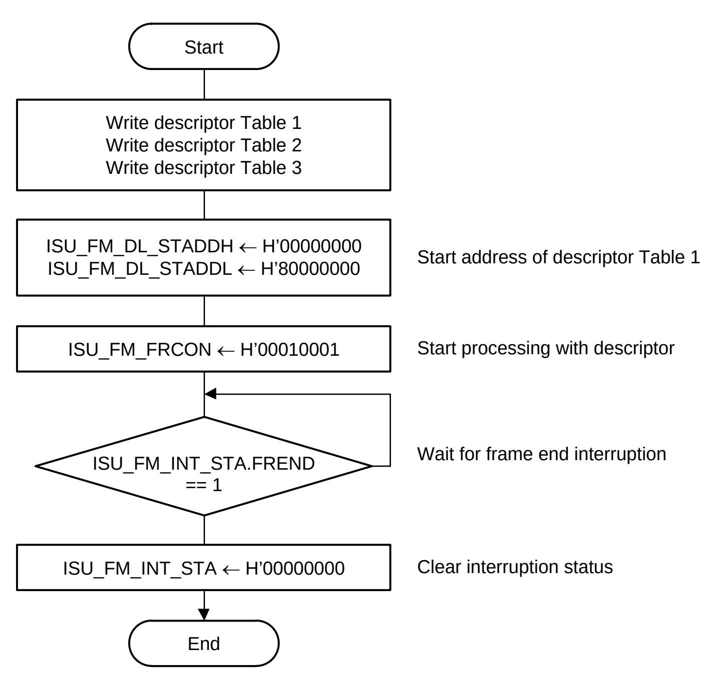


Figure 36.49 Flowchart of Image Size Conversion Example 2

36.5.2.3 NTSC (720 × 480) → WQVGA (320 × 240)

Set as follows to handle image size conversion.

Table 36.20 Image Size Conversion Example 3

Item	Description
Assuming that the following image is input from the CRU	
Input size	720 × 480
Input format	YCbCr422 8-bit UYVY, Interleave * For IP conversion, see **Section 36.5.1**.
Input start address	H’0_5000_0000 (InterlaceTop) H’0_5010_0000 (InterlaceBottom) H’0_5020_0000 (Progressive)
Input stride	Plane 0 = H’5A0
ISU setting	
Output (CROP) size	320 × 240
Output format	YCbCr422 8-bit UYVY, Interleave
Output start address	Plane 0 = H’0_6000_000
Output stride	Plane 0 = H’400
Processing method	Register
Color format conversion	None

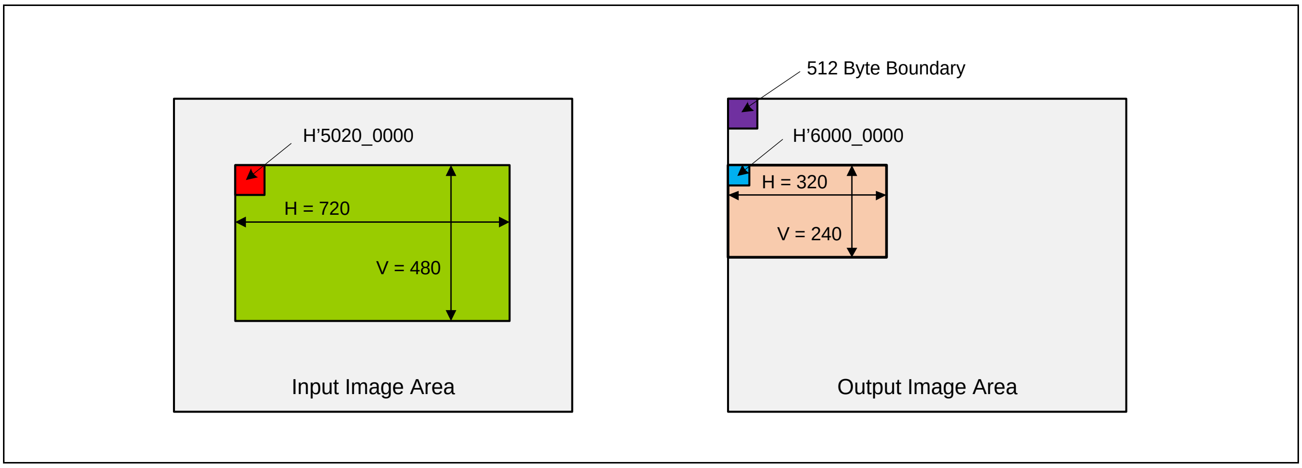


Figure 36.50 Image Size Conversion Example 3

Obtain the multiplication coefficient in the horizontal direction.

720 / 320 = 2.25
HMANT (integer part) = 2 = H’2
HFRAC (decimal part) = 0.25 × 4096 (Q12) = 1024 = H’400

Obtain the multiplication coefficient in the vertical direction.

480 / 240 = 2.0
VMANT (integer part) = 2 = H’2
VFRAC (decimal part) = 0.0 × 4096 (Q12) = 0.0 = H’0

Note: HFRAC and VFRAC are set by unsigned fixed point Q12 and are truncated after the decimal point.

Table 36.21 Image Size Conversion Example 3 (Register Settings 1, InterlaceTop Image)

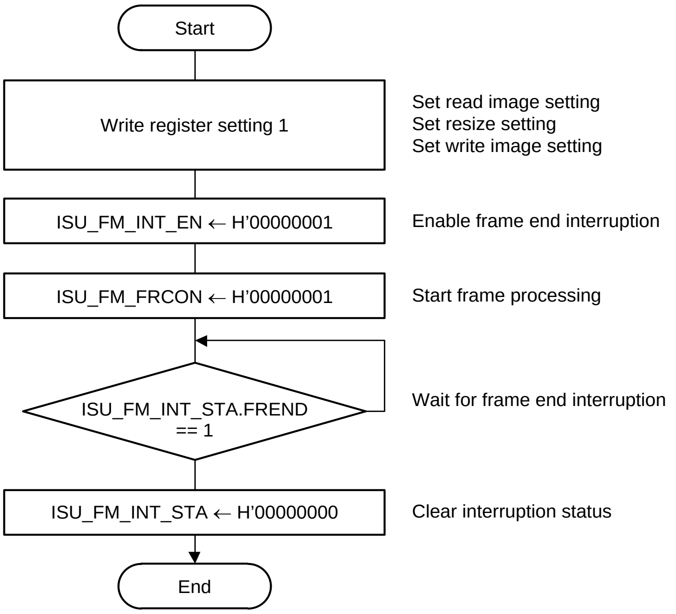
Register	Set Value	Comment
ISU_RPF_SRC_SIZE	H’02D000F0	Input size H = 720, V = 240
ISU_RPF_SRC_STRD	H’05A00000	Input stride 720*2 (pixel/byte) = 1440
ISU_RPF_SRC_ADDH_PL0	H’00000000	Upper 3 bits of Plane 0 input start address H’0
ISU_RPF_SRC_ADDL_PL0	H’50000000	Lower 32 bits of Plane 0 input start address H’5000_0000
ISU_RPF_FMT	H’00000020	Input format YCbCr422 8-bit UYVY (Interleave)
ISU_RS_HSCALE	H’00010000	Multiplication coefficient in the horizontal direction HMANT = H’1, HFRAC = H’000
ISU_RS_VSCALE	H’00010000	Multiplication coefficient in the vertical direction VMANT = H’1, VFRAC = H’000
ISU_RS_OS_CROP	H’02D000F0	CROP size H = 720, V = 240
ISU_WPF_DST_ADDH_PL0	H’00000000	Upper 3 bits of Plane 0 output start address H’0
ISU_WPF_DST_ADDL_PL0	H’50200000	Lower 32 bits of Plane 0 output start address H’5020_0000
ISU_WPF_DST_STRD	H’0C000000	Output stride H’0C00
ISU_WPF_FMT	H’00000020	Output format YCbCr422 8-bit UYVY (Interleave)
ISU_WPF_MUL1	H’00000400	K11 = H’400
ISU_WPF_MUL4	H’04000000	K22 = H’400
ISU_WPF_MUL6	H’00000400	K33 = H’400
ISU_WPF_CLP1	H’0000FF00	CLP (MAX/MIN)_A = H’FF, H’00
ISU_WPF_CLP2	H’FF00FF00	CLP (MAX/MIN)_B = H’FF, H’00, C = H’FF, H’00

Table 36.22 Image Size Conversion Example 3 (Register Settings 2, InterlaceBottom Image)

Register	Set Value	Comment
ISU_RPF_SRC_ADDL_PL0	H’50100000	Lower 32 bits of Plane 0 input start address H’5020_0000
ISU_WPF_DST_ADDL_PL0	H’50200600	Lower 32 bits of Plane 0 input start address H’5020_0600

Table 36.23 Image Size Conversion Example 3 (Register Settings 3, Progressive Image)

Register	Set Value	Comment
ISU_RPF_SRC_SIZE	H’02D001E0	Input size H = 720, V = 480
ISU_RPF_SRC_STRD	H’06000000	Input stride H’600
ISU_RPF_SRC_ADDL_PL0	H’50200000	Lower 32 bits of Plane 0 input start address H’5020_0000
ISU_RS_HSCALE	H’00020400	Multiplication coefficient in the horizontal direction HMANT = H’2, HFRAC = H’400
ISU_RS_VSCALE	H’00020000	Multiplication coefficient in the vertical direction VMANT = H’2, VFRAC = H’000
ISU_RS_OS_CROP	H’014000F0	CROP size H = 320, V = 240
ISU_WPF_DST_ADDL_PL0	H’60000000	Lower 32 bits of Plane 0 output start address H’6000_0000
ISU_WPF_DST_STRD	H’04000000	Output stride H’0400



Note: Repeat same operations for register setting 2 & 3.

Figure 36.51 Flowchart of Image Size Conversion Example 3

36.5.2.4 5M (2800 × 2047) → FullHD (1920 × 1080)

Set as follows to handle image size conversion.

Table 36.24 Image Size Conversion Example 4

Item	Description
Assuming that the following image is input from the CRU	
Input size	2800 × 2047
Input format	ARGB8888 * (The ISU swaps data because of the BGRA format.)
Input start address	H’0_5000_0000
Input stride	Plane0 = H’2BC0
ISU setting	
Output (CROP) size	1920 × 1080
Output format	ARGB8888
Output start address	Plane0 = H’0_6000_000
Output stride	Plane0 = H’1E00
Processing method	Register
Color format conversion	None

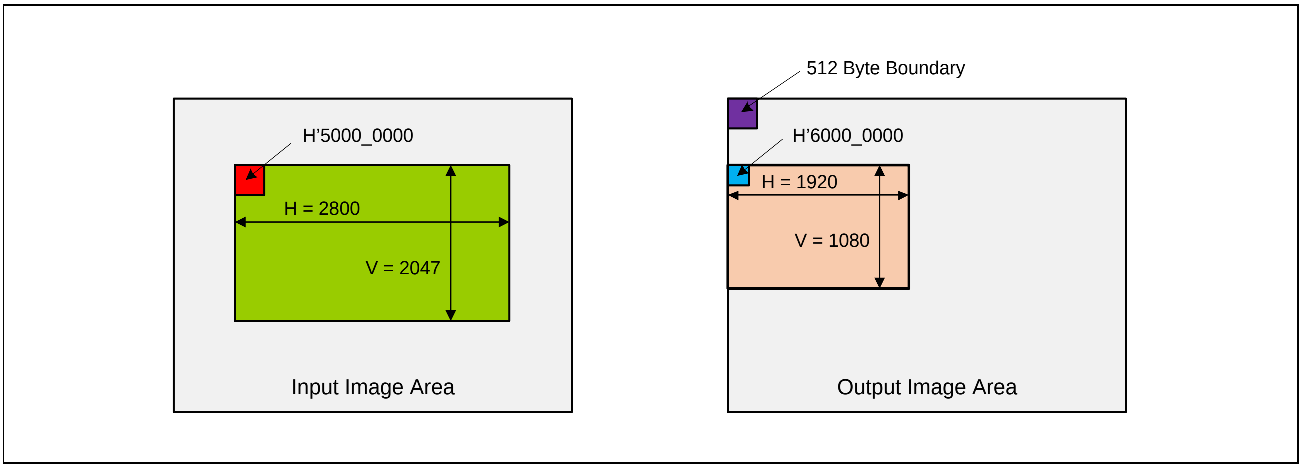


Figure 36.52 Image Size Conversion Example 4

Obtain the multiplication coefficient in the horizontal direction/

2800 / 1920 = 1.458333...
HMANT (integer part) = 1 = H’1
HFRAC (decimal part) = 0.458333 × 4096 (Q12) = 1877.3320 ... = H’755

Obtain the multiplication coefficient in the vertical direction.

2047 / 1080 = 1.895370...
VMANT (integer part) = 1 = H’1
VFRAC (decimal part) = 0.895370 × 4096 (Q12) = 3667.43552... = H’E53

Note: HFRAC and VFRAC are set by unsigned fixed point Q12 and are truncated after the decimal point.

Table 36.25 Image Size Conversion Example 4 (Register Settings)

Register	Set Value	Comment
ISU_RPF_SRC_SIZE	H'0AF007FF	Input size H = 2800, V = 2047
ISU_RPF_SRC_STRD	H'2BC00000	Input stride PL0 = H'2BC0
ISU_RPF_SRC_ADDH_PL0	H'00000000	Upper 3 bits of Plane 0 input start address H'0
ISU_RPF_SRC_ADDL_PL0	H'50000000	Lower 32 bits of Plane 0 input start address H'5000_0000
ISU_RPF_FMT	H'00000005	Input format ARGB8888
ISU_RPF_SRC_DSWAP	H'00000003	Data swap 3
ISU_RS_HSCALE	H'00010755	Multiplication coefficient in the horizontal direction HMANT = H'1, HFRAC = H'755
ISU_RS_VSCALE	H'00010E53	Multiplication coefficient in the vertical direction VMANT = H'1, VFRAC = H'E35
ISU_RS_OS_CROP	H'07800438	CROP size H = 1920, V = 1080
ISU_WPF_DST_ADDH_PL0	H'00000000	Upper 3 bits of Plane 0 output start address H'0
ISU_WPF_DST_ADDL_PL0	H'60000000	Lower 32 bits of Plane 0 output start address H'6000_0000
ISU_WPF_DST_STRD	H'1E000000	Output stride PL0 = H'1E00
ISU_WPF_FMT	H'00000005	Output format ARGB8888
ISU_WPF_MUL1	H'00000400	K11 = H'400
ISU_WPF_MUL4	H'04000000	K22 = H'400
ISU_WPF_MUL6	H'00000400	K33 = H'400
ISU_WPF_CLP1	H'0000FF00	CLP (MAX/MIN)_A = H'FF, H'00
ISU_WPF_CLP2	H'FF00FF00	CLP (MAX/MIN)_B = H'FF, H'00, C = H'FF, H'00

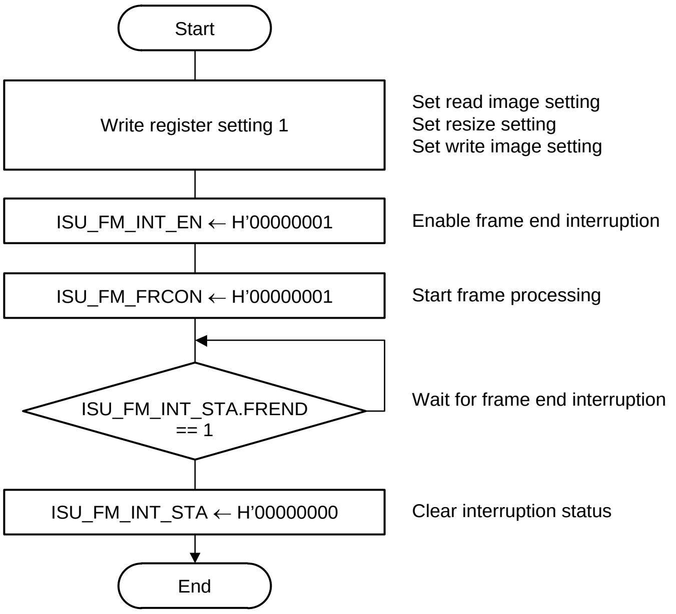


Figure 36.53 Flowchart of Image Size Conversion Example 4

36.5.3 Color Format Conversion

36.5.3.1 YCbCr422 → YCbCr420

Set as follows to handle color format conversion.

Table 36.26 Color Format Conversion Example 1

Item	Description
Assuming that the following image is input from the CRU	
Input size	320 × 240
Input format	YCbCr422 8-bit UYVY (Interleave)
Input start address	0_5000_0000
Input stride	Plane 0 = H’280
ISU setting	
Output (CROP) size	320 × 240
Output format	YCbCr420 8-bit NV12 (Semi-Planar)
Output start address	Plane 0 = H’0_6000_000 Plane 1 = H’0_6002_000
Output stride	Plane 0 = H’140 Plane 1 = H’140
Processing method	Register
Color format conversion	None

Table 36.27 Color Format Conversion Example 1 (Register Settings)

Register	Set Value	Comment
ISU_RPF_SRC_SIZE	H’014000F0	Input size H = 320, V = 240
ISU_RPF_SRC_STRD	H’02800000	Input stride PL0 = H’0280
ISU_RPF_SRC_ADDH_PL0	H’00000000	Upper 3 bits of Plane 0 input start address H’0
ISU_RPF_SRC_ADDL_PL0	H’50000000	Lower 32 bits of Plane 0 input start address H’5000_0000
ISU_RPF_FMT	H’00000020	Input format YCbCr422 8-bit UYVY (Interleave)
ISU_RS_HSCALE	H’00010000	Multiplication coefficient in the horizontal direction HMANT = H’1, HFRAC = H’000
ISU_RS_VSCALE	H’00010000	Multiplication coefficient in the vertical direction VMANT = H’1, VFRAC = H’000
ISU_RS_OS_CROP	H’014000F0	CROP size H = 320, V = 240
ISU_WPF_DST_ADDH_PL0	H’00000000	Upper 3 bits of Plane 0 output start address H’0
ISU_WPF_DST_ADDL_PL0	H’60000000	Lower 32 bits of Plane 0 output start address H’6000_0000
ISU_WPF_DST_ADDH_PL1	H’00000000	Upper 3 bits of Plane 1 output start address H’0
ISU_WPF_DST_ADDL_PL1	H’60020000	Lower 32 bits of Plane 1 output start address H’6002_0000
ISU_WPF_DST_STRD	H’01400140	Output stride PL0 = H’0140, PL1 = H’0140
ISU_WPF_FMT	H’00000023	Output format YCbCr420 8-bit NV12 (Semi-Planar)
ISU_WPF_MUL1	H’00000400	K11 = H’400
ISU_WPF_MUL4	H’04000000	K22 = H’400
ISU_WPF_MUL6	H’00000400	K33 = H’400
ISU_WPF_CLP1	H’0000FF00	CLP (MAX/MIN)_A = H’FF, H’00
ISU_WPF_CLP2	H’FF00FF00	CLP (MAX/MIN)_B = H’FF, H’00, C = H’FF, H’00

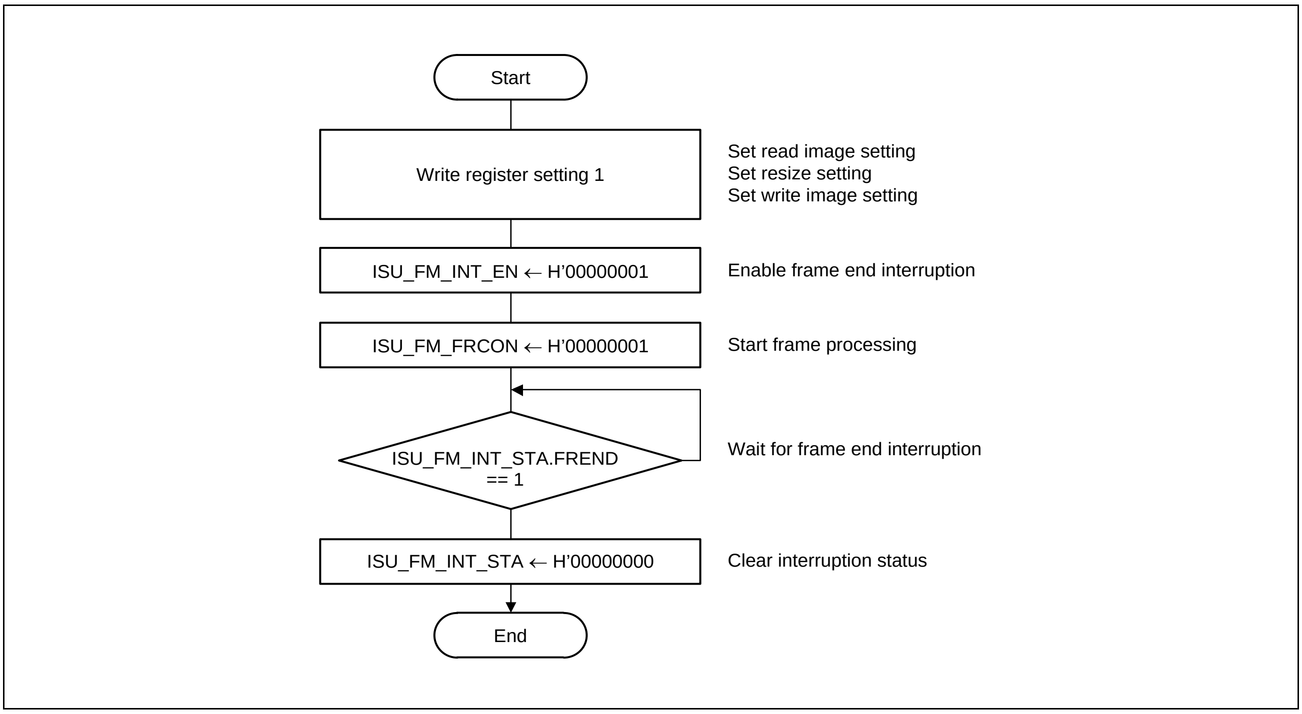


Figure 36.54 Flowchart of Color Format Conversion 1

36.5.3.2 YCbCr422 → ARGB8888

Set as follows to handle color format conversion.

Table 36.28 Color Format Conversion Example 2

Item	Description
Assuming that the following image is input from the CRU	
Input size	320 × 240
Input format	YCbCr422 8-bit UYVY (Interleave)
Input start address	H’0_5000_0000
Input stride	Plane 0 = H’280
ISU setting	
Output (CROP) size	320 × 240
Output format	ARGB8888
Output start address	Plane 0 = H’0_6000_000
Output stride	Plane 0 = H’500
Processing method	Register
Color format conversion	YCbCr → RGB (BT.601 SDTV)

The following shows the matrix operation formula for conversion and color correction.

$$\begin{bmatrix} C1' \\ C2' \\ C3' \end{bmatrix} = \begin{bmatrix} K11 & K12 & K13 \\ K21 & K22 & K23 \\ K31 & K32 & K33 \end{bmatrix} \cdot \begin{bmatrix} A - OFS\_A1 \\ B - OFS\_B1 \\ C - OFS\_C1 \end{bmatrix} + \begin{bmatrix} OFST\_A2 \\ OFST\_B2 \\ OFST\_C2 \end{bmatrix}$$

$$C1'' = MAX\,(clip\_a\_min,\ Min\,(clip\_a\_max,\ C1'))$$
$$C2'' = MAX\,(clip\_b\_min,\ Min\,(clip\_b\_max,\ C2'))$$
$$C3'' = MAX\,(clip\_c\_min,\ Min\,(clip\_c\_max,\ C3'))$$

Figure 36.55 Matrix Operation Formula for Conversion and Color Correction

The following shows the YCbCr422-to-RGB conversion formula (BT.601 SDTV).

$R = 1.164\ (Y - 16) + 1.596\ (Cr - 128)$
$G = 1.164\ (Y - 16) - 0.391\ (Cb - 128) - 0.813\ (Cr - 128)$
$B = 1.164\ (Y - 16) + 2.018\ (Cb - 128)$
* R/G/B: 0 to 255

Perform YCbCr-to-RGB conversion with the following allocation.

$R = K11\ (Y - OFST\_A1) + K12\ (Cb - OFST\_B1) + K13\ (Cr - OFST\_C1) + OFST\_A2$
$G = K21\ (Y - OFST\_A1) + K22\ (Cb - OFST\_B1) + K23\ (Cr - OFST\_C1) + OFST\_B2$
$B = K31\ (Y - OFST\_A1) + K32\ (Cb - OFST\_B1) + K33\ (Cr - OFST\_C1) + OFST\_C2$

Note: K** is set by signed fixed point Q10.
Example: K11 = 1.164 × 1024 = 1191.936 = H’4A8
Rounded to the nearest whole number.

Table 36.29 Color Format Conversion Example 2 (Register Settings)

Register	Set Value	Comment
ISU_RPF_SRC_SIZE	H'014000F0	Input size H = 320, V = 240
ISU_RPF_SRC_STRD	H'02800000	Input stride PL0 = H'0280
ISU_RPF_SRC_ADDH_PL0	H'00000000	Upper 3 bits of Plane 0 input start address H'0
ISU_RPF_SRC_ADDL_PL0	H'50000000	Lower 32 bits of Plane 0 input start address H'5000_0000
ISU_RPF_FMT	H'00000020	Input format YCbCr422 8-bit UYVY (Interleave)
ISU_RS_HSCALE	H'00010000	Multiplication coefficient in the horizontal direction HMANT = H'1, HFRAC = H'000
ISU_RS_VSCALE	H'00010000	Multiplication coefficient in the vertical direction VMANT = H'1, VFRAC = H'000
ISU_RS_OS_CROP	H'014000F0	CROP size H = 320, V = 240
ISU_WPF_DST_ADDH_PL0	H'00000000	Upper 3 bits of Plane 0 output start address H'0
ISU_WPF_DST_ADDL_PL0	H'60000000	Lower 32 bits of Plane 0 output start address H'6000_0000
ISU_WPF_DST_STRD	H'05000000	Output stride PL0 = H'0500
ISU_WPF_FMT	H'00000005	Output format ARGB8888
ISU_WPF_CCOL	H'00000002	With color conversion correction
ISU_WPF_MUL1	H'000004A8	K11 = H'04A8
ISU_WPF_MUL2	H'00000662	K12 = H'0000, K13 = H'0662
ISU_WPF_MUL3	H'000004A8	K21 = H'04A8
ISU_WPF_MUL4	H'3E703CBF	K22 = H'3E70, K23 = H'3CBF
ISU_WPF_MUL5	H'000004A8	K31 = H'04A8
ISU_WPF_MUL6	H'08120000	K32 = H'0812, K33 = H'0000
ISU_WPF_OFST1	H'10808000	OFST_A1 = H'10, B1 = H'80, C1 = H'80
ISU_WPF_OFST2	H'00000000	OFST_A2 = H'00, B2 = H'00, C2 = H'00
ISU_WPF_CLP1	H'0000FF00	CLP (MAX/MIN)_A = H'FF, H'00
ISU_WPF_CLP2	H'FF00FF00	CLP (MAX/MIN)_B = H'FF, H'00, C = H'FF, H'00

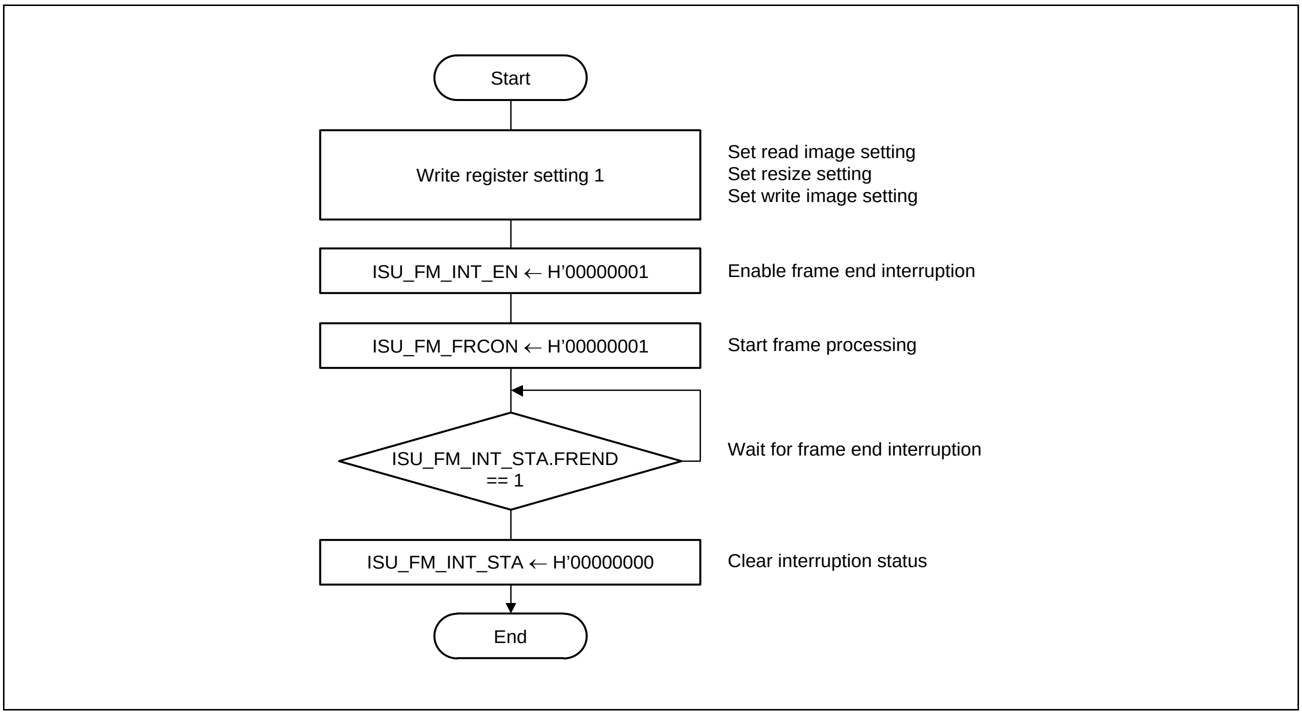


Figure 36.56 Flowchart of Color Format Conversion 2

36.5.3.3 BGR888 → YCbCr422

Set as follows to handle color format conversion.

Table 36.30 Color Format Conversion Example 3

Item	Description
Assuming that the following image is input from the CRU	
Input size	320 × 240
Input format	BGR888
Input start address	H’0_5000_0000
Input stride	Plane 0 = H’03C0
ISU setting	
Output (CROP) size	320 × 240
Output format	YCbCr422 8-bit UYVY (Interleave)
Output start address	Plane 0 = H’0_6000_000
Output stride	Plane 0 = H’0280
Processing method	Register
Reduction setting	No reduction
Color format conversion	RGB → YCbCr (BT.601 SDTV)

The following shows the matrix operation formula for conversion and color correction.

$$\begin{bmatrix} C1' \\ C2' \\ C3' \end{bmatrix} = \begin{bmatrix} K11 & K12 & K13 \\ K21 & K22 & K23 \\ K31 & K32 & K33 \end{bmatrix} \cdot \begin{bmatrix} A - OFS\_A1 \\ B - OFS\_B1 \\ C - OFS\_C1 \end{bmatrix} + \begin{bmatrix} OFST\_A2 \\ OFST\_B2 \\ OFST\_C2 \end{bmatrix}$$

$$C1'' = MAX\ (clip\_a\_min,\ Min\ (clip\_a\_max,\ C1'))$$
$$C2'' = MAX\ (clip\_b\_min,\ Min\ (clip\_b\_max,\ C2'))$$
$$C3'' = MAX\ (clip\_c\_min,\ Min\ (clip\_c\_max,\ C3'))$$

Figure 36.57 Matrix Operation Formula for Conversion and Color Correction

The following shows the RGB-to-YCbCr422 conversion formula (BT.601 SDTV).

$Y = 0.257R + 0.504G + 0.098B + 16$
$Cb = -0.148R - 0.291G + 0.439B + 128$
$Cr = 0.439R - 0.368G - 0.071B + 128$
* Brightness: 16 to 235
* Color difference: 16 to 240

Perform RGB-to-YCbCr conversion with the following allocation.

$Y = K11\ (R - OFST\_A1) + K12\ (G - OFST\_B1) + K13\ (B - OFST\_C1) + OFST\_A2$
$Cb = K21\ (R - OFST\_A1) + K22\ (G - OFST\_B1) + K23\ (B - OFST\_C1) + OFST\_B2$
$Cr = K31\ (R - OFST\_A1) + K32\ (G - OFST\_B1) + K33\ (B - OFST\_C1) + OFST\_C2$

Note: K** is set by signed fixed point Q10.
Example: K11 = 0.257 × 1024 = 263.168 = H’107
Rounded to the nearest whole number.

Table 36.31 Color Format Conversion Example 3 (Register Settings)

Register	Set Value	Comment
ISU_RPF_SRC_SIZE	H'014000F0	Input size H = 320, V = 240
ISU_RPF_SRC_STRD	H'03C00000	Input stride PL0 = H'03C0
ISU_RPF_SRC_ADDH_PL0	H'00000000	Upper 3 bits of Plane 0 input start address H'0
ISU_RPF_SRC_ADDL_PL0	H'50000000	Lower 32 bits of Plane 0 input start address H'5000_0000
ISU_RPF_FMT	H'00000004	Input format BGR888
ISU_RS_HSCALE	H'00010000	Multiplication coefficient in the horizontal direction HMANT = H'1, HFRAC = H'000
ISU_RS_VSCALE	H'00010000	Multiplication coefficient in the vertical direction VMANT = H'1, VFRAC = H'000
ISU_RS_OS_CROP	H'014000F0	CROP size H = 320, V = 240
ISU_WPF_DST_ADDH_PL0	H'00000000	Upper 3 bits of Plane 0 output start address H'0
ISU_WPF_DST_ADDL_PL0	H'60000000	Lower 32 bits of Plane 0 output start address H'6000_0000
ISU_WPF_DST_STRD	H'02800000	Output stride PL0 = H'0280
ISU_WPF_FMT	H'00000020	Output format YCbCr422 8-bit UYVY (Interleave)
ISU_WPF_CCOL	H'00000002	With color conversion correction
ISU_WPF_MUL1	H'00000107	K11 = H'0107
ISU_WPF_MUL2	H'02040064	K12 = H'0204, K13 = H'0064
ISU_WPF_MUL3	H'00003F68	K21 = H'3F68
ISU_WPF_MUL4	H'3ED601C2	K22 = H'3ED6, K23 = H'01C2
ISU_WPF_MUL5	H'000001C2	K31 = H'01C2
ISU_WPF_MUL6	H'3E873FB7	K32 = H'3E87, K33 = H'3FB7
ISU_WPF_OFST1	H'00000000	OFST_A1 = H'00, B1 = H'00, C1 = H'00
ISU_WPF_OFST2	H'10808000	OFST_A2 = H'10, B2 = H'80, C2 = H'80
ISU_WPF_CLP1	H'0000EB10	CLP (MAX/MIN)_A = H'EB, H'10
ISU_WPF_CLP2	H'F010F010	CLP (MAX/MIN)_B = H'F0, H'10, C = H'F0, H'10

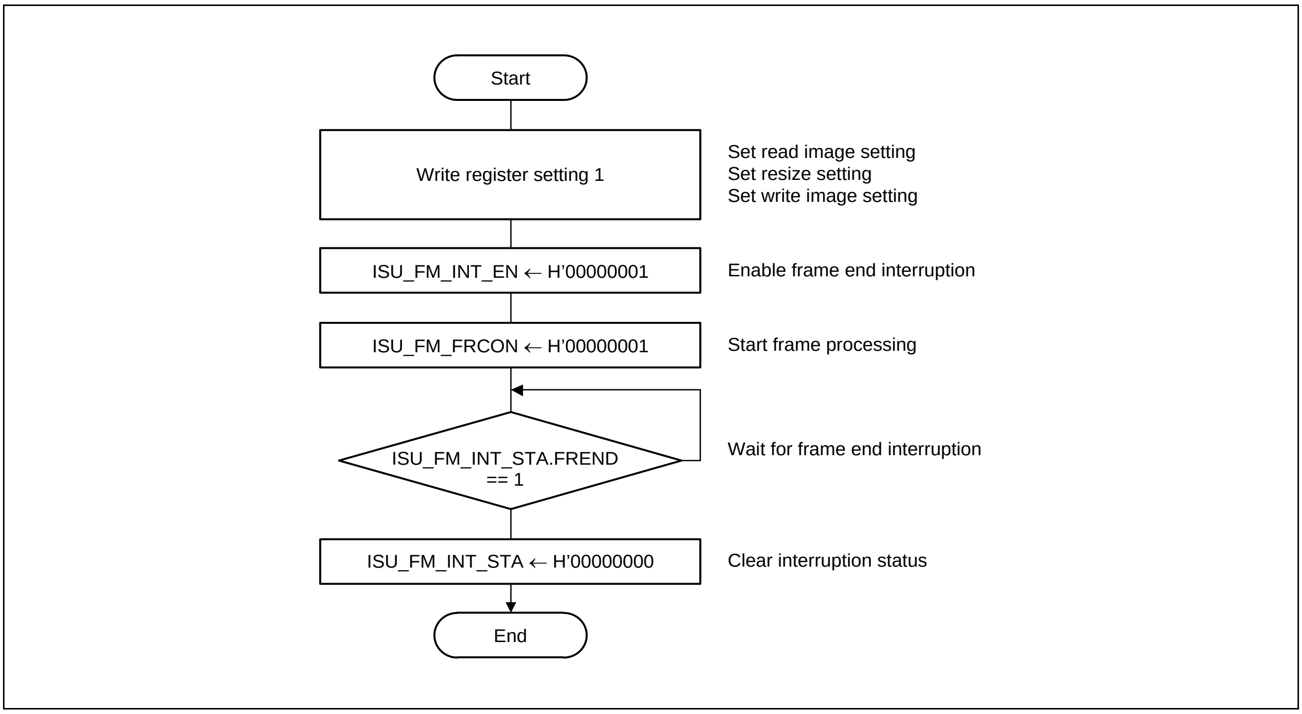


Figure 36.58 Flowchart of Color Format Conversion Example 3

37. Video Codec Processor (VCPL4)

The VCPL4 is a H.264 codec module. Require software or the library to RENESAS for operating this module.

Figure 37.1 shows VCPL4 and related modules for this LSI. The VCPL4 is connected memory bus (AXI-bus) via companion modules, FCPCS. For detail information, refer **Section 38**.

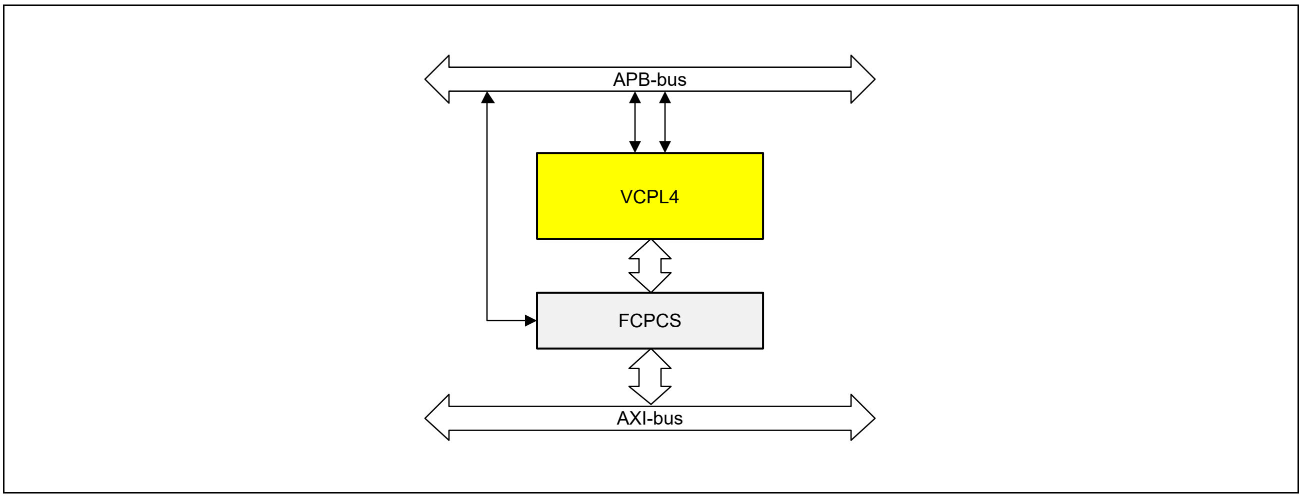


Figure 37.1 VCPL4 and Related Modules

37.1 Features

The VCPL4 has the following features.

- Support for H.264 decoding and encoding
- Support up to 1920 × 1080 resolution
- Data handling on a picture-by-picture basis
    - Encode/decode data one picture (frame or field) at a time.
- High picture quality
    - Support the H.264 high-efficiency coding tools (CABAC, 8 × 8 frequency conversion, and quantization matrix).
    - High-efficiency motion vector detection by a combination of discrete search and trace search
    - Optimal-mode selection by Rate-Distortion (RD) cost evaluation
    - Picture quality control based on activity analysis results which match visual models

Table 37.1 Main Function of the VCPL4

Item	Description
Encoding standard supported	● H.264/AVC – High Profile@Level 4.1 – Main Profile@Level 4.1 – Baseline Profile@Level 4.1 (The arbitrary slice order, flexible macroblock order and redundant slice aren't supported)
Decoding standard supported	● H.264/AVC – High Profile@Level 4.1 – Main Profile@Level 4.1 – Baseline Profile@Level 4.1 (The arbitrary slice order, flexible macroblock order and redundant slice aren't supported)
Maximum pixel rate	1920 × 1080p × 30 fps
Maximum bit rate	200 Mbps × 1ch *Note*: a target bitrate can be set for each picture by using Renesas software.
Picture size supported in decoding	● Minimum size: Horizontal 80 pixels × vertical 80 lines ● Maximum size: 1920 pixels × vertical 1080 lines ● Horizontal size is a multiple of 2 pixels, and vertical size is a multiple of 2 lines. Additionally, 1080 × 1920 is supported.
Picture size supported in encoding	● Minimum size: Horizontal 80 pixels × vertical 80 lines ● Maximum size: Horizontal 1920 pixels × vertical 1080 lines ● Horizontal size is a multiple of 8 pixels, and vertical size is a multiple of 8 lines. Additionally, 320 × 180, 854 × 480 (WVGA+), 1080 × 1920 are supported.
Color format	● YCbCr 4:2:0 semi-planar (Y and C plane) are supported as an input picture in encoding. ● YCbCr 4:2:0 semi-planar is supported as an output picture in decoding. ● Cb/Cr and Cr/Cb interleaving order is supported in YCbCr 4:2:0 semi-planar.
Bit stream format	● Video elementary stream as shown in **Figure 37.1**. ● Function to add or remove the emulation prevention byte in the H.264 bit stream is supported.
Picture structure	Frame structure and field structure

Table 37.2 Main Function of the VCPL4

Standard	Standard	Decoding Range
H.264	Slice header and subsequent data	Slice header (other than the first slice header in picture) and subsequent data

37.2 Usage Note

37.2.1 Limitations on Software Reset and Module Standby

Before executing following operations for VCPL4, confirm that VCPL4 is not encoding or decoding a picture. If VCPL4 is encoding or decoding a picture, wait completion of encoding or decoding.

- Execute Software Reset by CPG
- Enter Module Standby (stop clock supply) by CPG

37.2.2 Note on Input Bitstream for Decoding

In decoding process of VCPL4, input bitstream not conformant to video codec standard might cause hang of VCPL4.

37.3 Interrupt Request

VCPL4 interrupt requests are listed in the following table.

Table 37.3 Interrupt Requests

Unit Interrupt Signal	Outline
VCP.vcpl4.vint	Interruption request signal from VLC block
VCP.vcpl4.cint	Interruption request signal from CE block
VCP.vcpl4.vedcint	EDC error Interrupt from VLC block
VCP.vcpl4.cedcint	EDC error Interrupt from CE block

38. Frame Compression Processor (FCPCS)

The FCPCS is a companion module of a video processing module with Renesas lossless data compression and decompression (FCL) for the Renesas video codec processor (VCPL4). It provides data compression and decompression, data caching, and converting of AXI transaction in order to reduce the memory bandwidth.

The FCPCS is connected to codec IP cores called VCPL4 (H.264 decoder and encoder).

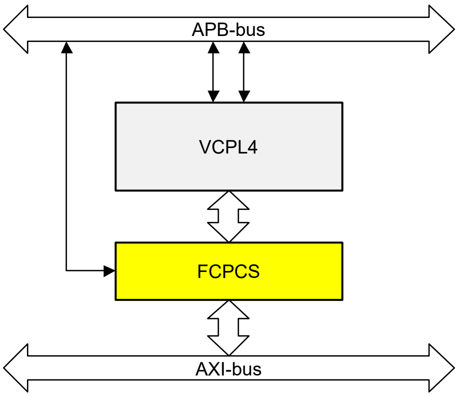


Figure 38.1 FCPCS and Related Modules

38.1 Features

The FCPCS has the following features.

- Support for Lossless compression for decoded picture and decompression for reference picture
- Support 1920 × 1080 resolution
- Support for reference picture cache
- Lossless compression ratio is typically 50%. It may not be compressed since Lossless compression relies on the statistical nature of the original image.

Table 38.1 Main Function of the FCPCS

Item	Description
Data compression and decompression	Renesas lossless (FCL) compression and decompression, Compression for decoded picture, and decompression for reference picture YCbCr 4:2:0 semi-planar and 8-bit depth are supported
Reference picture cache	64 Kbyte
AXI transaction	[Read access] • Out-of-order completion allowed. • Outstanding transactions supported. [Write access] • Out-of-order completion allowed. • Outstanding transactions supported.

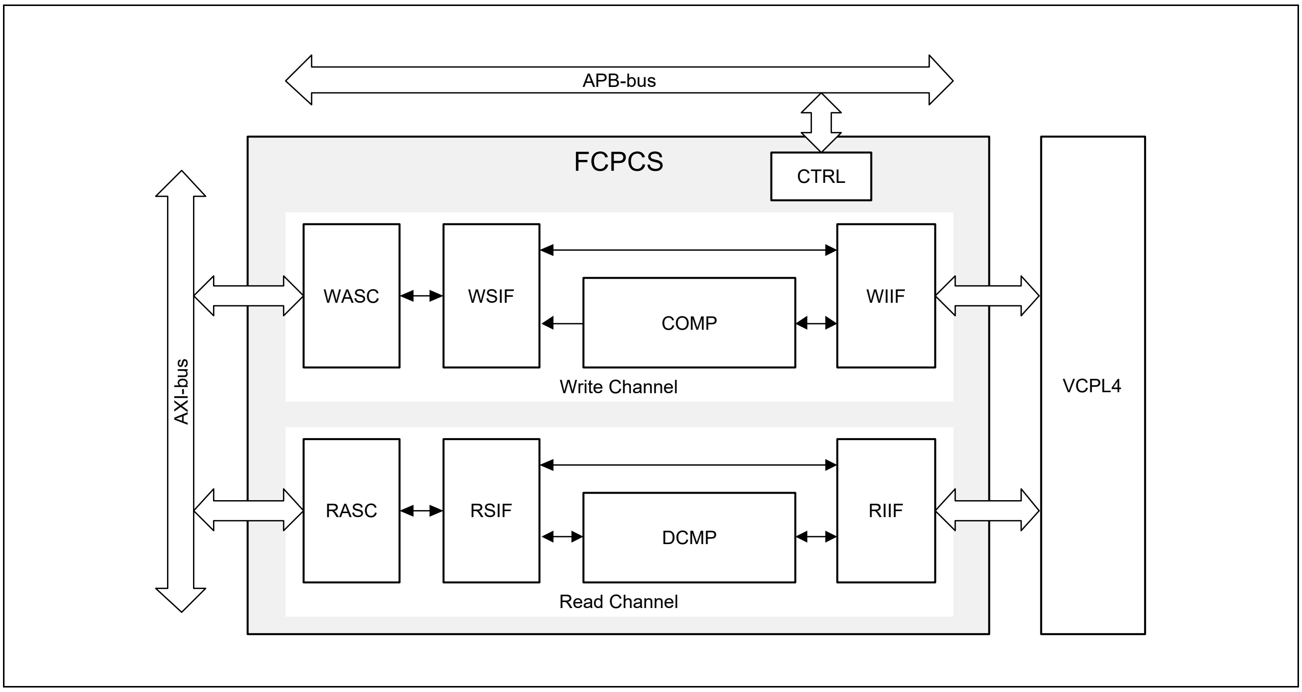


Figure 38.2 FCPCS Block Diagram

Table 38.2 Main Function of the FCPCS

Channel	Abbreviation	Description
Write Channel	COMP	Compression module
	WIIF	Write channel interface for VCPL4
	WSIF	System AXI write channel interface
	WASC	Synchronization of write channel
Read Channel	DCMP	Decompression and cache control module
	RIIF	Read channel interface for VCPL4
	RSIF	System AXI read channel interface
	RASC	Synchronization of read channel

38.2 Usage Note

38.2.1 Limitations on Software Reset and Module Standby

Before executing following operations for FCPCS, confirm that connected modules (VCPL4) are not encoding or decoding a picture. If connected modules are encoding or decoding a picture, wait completion of encoding and decoding.

- Execute Software Reset by CPG
- Enter Module Standby (stop clock supply) by CPG

38.3 Interrupt Request

FCPCS interrupt requests are listed in the following table.

Table 38.3 Interrupt Requests

Unit Interrupt Signal	Outline
VCP.fcpcs.edcint	EDC error Interrupt from FCPCS

39. 3D Graphics Engine (3DGE)

39.1 Overview

This LSI includes the Arm Mali-G31 GPU as a 3D Graphics Engine (3DGE). The Arm Mali-G31 GPU is a graphics acceleration platform that is based on open standards. It supports 2D graphics, 3D graphics, and General Purpose computing on GPU (GPGPU).

39.1.1 Features

The 3DGE has the following features.

- One single-pixel shader core
- 8Kbytes L2 cache
- A programmable architecture.
- An API feature set with support for shader-based and fixed-function graphics APIs.
- Anti-aliasing capabilities.
- An effective core for General Purpose computing on GPU (GPGPU) applications.
- High memory bandwidth and low power consumption for 3D graphics content.
- Latency tolerance.
- Compressed texture formats.
- Coherency aware interconnects for system memory and resource sharing.

The 3DGE and software support compute and graphics API standards.

The 3DGE supports these compute API standards:

- OpenCL 2.0 Full Profile.

The 3DGE supports these graphics API standards:

- OpenGLES 1.1, 2.0, 3.0, 3.1 and 3.2.

39.1.2 Block Diagram

The 3DGE contains the shader processor core and associated blocks, including the tiler, the memory-management unit, and the level 2 cache memory subsystem.

Figure 39.1 shows a block diagram.

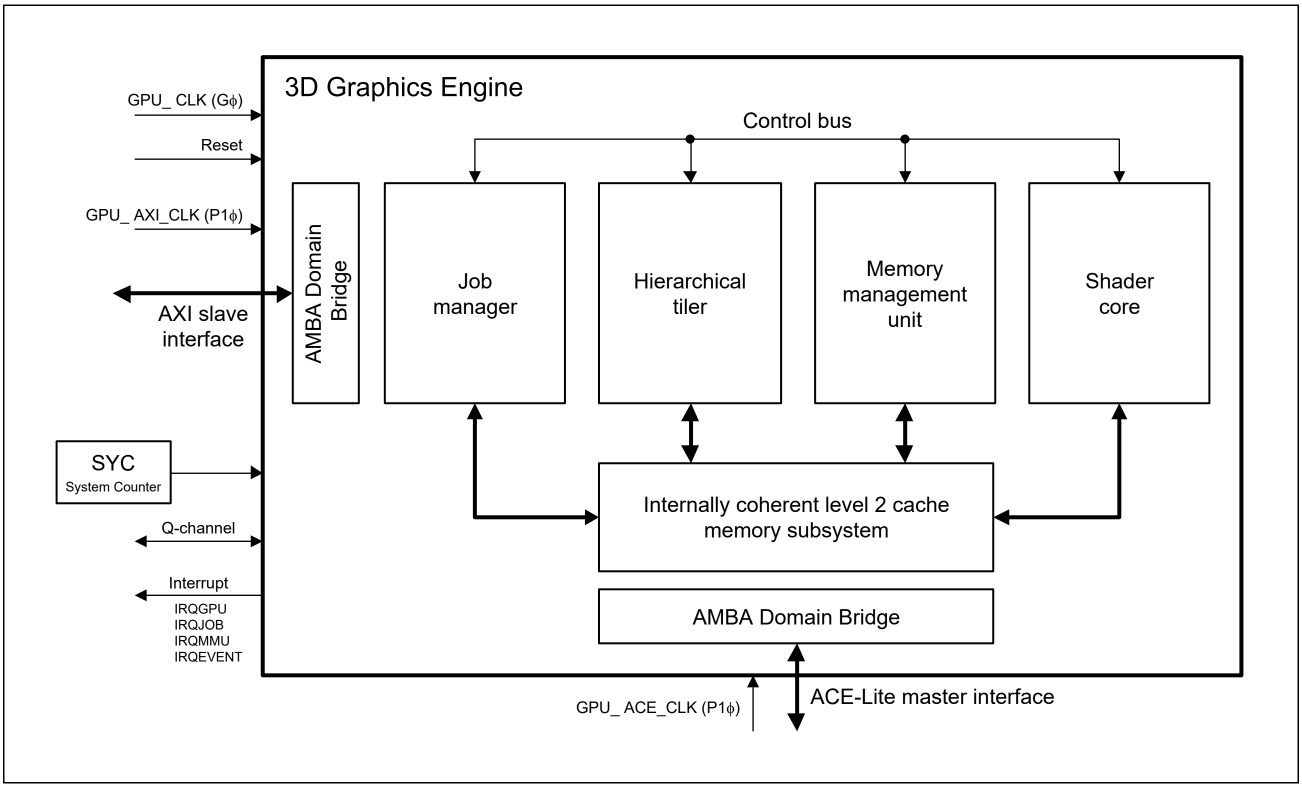


Figure 39.1 Block Diagram

The 3DGE uses a centralized hardware resource manager, called the job manager, that controls the internal GPU functions.

The application software runs on the application processor. When it requires graphics to be displayed on screen, it uses the graphics device driver to send a graphics job to the 3DGE. In the 3DGE, the job manager receives the job, interprets it, and then sends a series of graphics tasks to the internal functional units.

The job manager issues and sends commands to specific functional blocks in the 3DGE.

- The shader core is the main processing engines of the 3DGE. The shader core carries out all the rendering and computation operations.
- The tiler creates lists of all the objects in a scene, so that the shader core processes the objects efficiently.
- The memory management unit performs virtual address to physical address translation of external bus interface accesses.
- The Level 2 cache memory subsystem provides caching for all internal master blocks. It also supports cache coherency for caches in the shader cores. The L2 cache sits between the internal master blocks and the external interconnect. All transactions go through the cache. If the cache cannot service an internal transaction, it performs an external transaction to read or write data.

39.1.3 External Pins

In 3D Graphics Engine, there are no external pins.

39.1.4 Register Configuration

Base address of registers is allocated to H’0_1184_0000. The address range is 64 KB (H’0_1184 0000 - H’0_1184 FFFF).

39.1.5 Usage Note

Renesas does not provide technical support for users to program 3DGE registers directly. Renesas provides users with drivers to program 3DGE. 3DGE supports OpenGLES as a graphics API. When drawing graphics, control according to this API.

39.1.6 Starting SYC

When using 3DGE, it is necessary to supply the clock to the SYC (system counter) and release the reset to keep it running.

40. SD/MMC Host Interface

NOTE

Development of the SD host--related products needs the conclusion of the following agreement. SD Host/Ancillary Product License Agreement (SD HALA)

40.1 Overview

40.1.1 Features

- 2 channels
- Channel 0 supports SDHI / e-MMC
- Channel 1 supports SDHI
- SD memory/IO card interface (1-bit/4-bit SD bus)
- SD, SDHC, and SDXC SD memory card access supported
- Default, high-speed, UHS-I/SDR50, and SDR104 transfer modes supported
- SD clock (SD_CLK) frequency = SDxϕ/4 frequency/2^n (n = 0 to 9) (x = 0, 1)
- Error check function: CRC7 (for command/response), CRC16 (for data)
- Interrupt request: 2
- Card detect function
- Write protect supported
- MMC interface (1-/4-/8-bit MMC bus)
- e-MMC device access supported
- Backward-compatible, high-speed, HS200 transfer modes supported
- High-priority interrupt (HPI) supported

Block Diagram

Figure 40.1 shows a block diagram of the SD/MMC host interfaces.

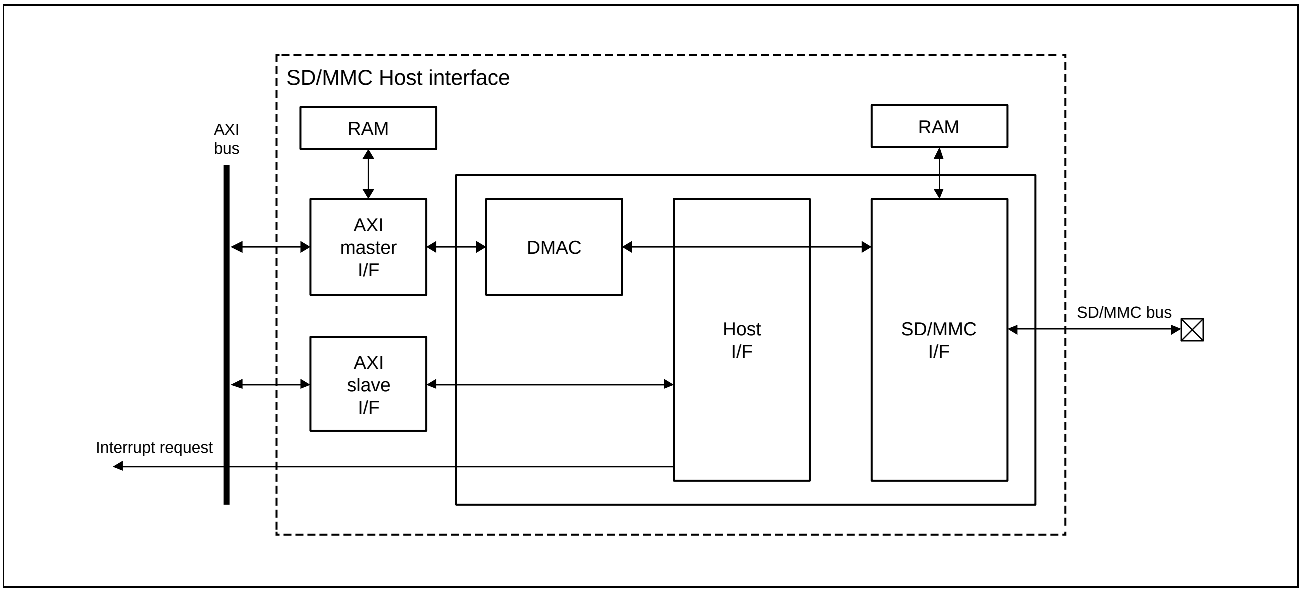


Figure 40.1 Block Diagram of SD/MMC Host Interface

40.1.2 External Pins

Table 40.1 lists the input and output pins used by the interface. The operating voltage on the pins of the SD/MMC host interface is 3.3 V or 1.8 V. Channels 0 and 1 can operate with different voltages. Before using the pins of the SD/MMC host interface, be sure to set the SD ch0/1 IO Voltage Mode Control Register (SD_ch0 and SD_ch1) and Driving Ability Control Register (IOLH).

For details, see the following sections:

Section 41.3.7, Driving Ability Control Register,
Section 41.3.17, SD ch0 IO Voltage Mode Control Register,
Section 41.3.18, SD ch1 IO Voltage Mode Control Register.

Table 40.1 Pin Configuration

PIN Name	I/O	Function
SDx_CLK*[1]	O	SD/MMC clock output
SDx_CMD*[1]	I/O	SD/MMC command output, response input
SDx_DATA0*[1]	I/O	SD/MMC Data 0 [bit 0]
SDx_DATA1*[1]	I/O	SD/MMC Data 1 [bit 1], SDIO interrupt
SDx_DATA2*[1]	I/O	SD/MMC Data 2 [bit 2], read wait
SDx_DATA3*[1]	I/O	SD/MMC Data 3 [bit 3], Card detection
SD0_DATA4*[1]	I/O	SD/MMC Data 4 [bit 4]
SD0_DATA5*[1]	I/O	SD/MMC Data 5 [bit 5]
SD0_DATA6*[1]	I/O	SD/MMC Data 5 [bit 6]
SD0_DATA7*[1]	I/O	SD/MMC Data 7 [bit 7]
SDx_CD*[1]	I	SD/MMC card detection*[2]
SDx_WP*[1]	I	SD/MMC write protection*[2]
SD0_RST#	O	SD/MMC reset

Note 1. x (= 0, 1) is the channel number of the SD/MMC host interface. In this manual, those pins are referred as SDDAT0, SDDAT1, …, SDDAT7.

Note 2. Fix to 1 when not in use.

40.1.3 Register Configuration

The base addresses for each channel are as follows. The SD interface and MMC interface are switched by setting the command type register (SD_CMD).

Channel 0:
H’0_11C0_0000 (Cortex-A55 Address Space)
H’41C0_0000 (Cortex-M33 Address Space Non-Secure)
H’51C0_0000 (Cortex-M33 Address Space Secure)

Channel 1:
H’0_11C1_0000 (Cortex-A55 Address Space)
H’41C1_0000 (Cortex-M33 Address Space Non-Secure)
H’51C1_0000 (Cortex-M33 Address Space Secure)

Table 40.2 Register Configurations (1/2)

Name	Abbreviation	Address	Access Width	Mirror
Command type register	SD_CMD	H’0000	16/32/64	
Command argument registers	SD_ARG	H’0010	16/32/64	
	SD_ARG1	H’0018	16/32/64	SD_ARG[31:16]
Data stop register	SD_STOP	H’0020	16/32/64	
Block count register	SD_SECCNT	H’0028	16/32/64	
Card response registers	SD_RSP10	H’0030	16/32/64	
	SD_RSP1	H’0038	16/32/64	SD_RSP10[31:16]
	SD_RSP32	H’0040	16/32/64	SD_RSP10[63:32]
	SD_RSP3	H’0048	16/32/64	SD_RSP32[31:16]
	SD_RSP54	H’0050	16/32/64	
	SD_RSP5	H’0058	16/32/64	SD_RSP54[31:16]
	SD_RSP76	H’0060	16/32/64	SD_RSP54[63:32]
	SD_RSP7	H’0068	16/32/64	SD_RSP76[31:16]
SD card interrupt flag register 1	SD_INFO1	H’0070	16/32/64	
SD card interrupt flag register 2	SD_INFO2	H’0078	16/32/64	
SD_INFO1 interrupt mask register	SD_INFO1_MASK	H’0080	16/32/64	
SD_INFO2 interrupt mask register	SD_INFO2_MASK	H’0088	16/32/64	
SD clock control register	SD_CLK_CTRL	H’0090	16/32/64	
Transfer data length register	SD_SIZE	H’0098	16/32/64	
Card access control option register	SD_OPTION	H’00A0	16/32/64	
SD error status register 1	SD_ERR_STS1	H’00B0	16/32/64	
SD error status register 2	SD_ERR_STS2	H’00B8	16/32/64	
SD buffer read/write register	SD_BUF0	H’00C0	16/32/64	
SDIO mode control register	SDIO_MODE	H’00D0	16/32/64	
SDIO interrupt flag register	SDIO_INFO1	H’00D8	16/32/64	
SDIO_INFO1 interrupt mask register	SDIO_INFO1_MASK	H’00E0	16/32/64	
DMA mode enable register	CC_EXT_MODE	H’0360	16/32/64	
Software reset register	SOFT_RST	H’0380	16/32/64	
Version register	VERSION	H’0388	16/32/64	
Host interface mode setting register	HOST_MODE	H’0390	16/32/64	
SD interface mode setting register	SDIF_MODE	H’0398	16/32/64	

Table 40.2 Register Configurations (2/2)

Name	Abbreviation	Address	Access Width	Mirror
SD status register*[1]	SD_STATUS	H’03C8	16/32/64	
DMAC mode register	DM_CM_DTRAN_MODE	H’0820	16/32/64	
DMAC control register	DM_CM_DTRAN_CTRL	H’0828	16/32/64	
DMAC reset register	DM_CM_RST	H’0830	16/32/64	
DMAC interrupt register 1	DM_CM_INFO1	H’0840	16/32/64	
DM_CM_INFO1 interrupt mask register	DM_CM_INFO1_MASK	H’0848	16/32/64	
DMAC interrupt register 2	DM_CM_INFO2	H’0850	16/32/64	
DM_CM_INFO2 interrupt mask register	DM_CM_INFO2_MASK	H’0858	16/32/64	
DMAC address register	DM_DTRAN_ADDR	H’0880	16/32/64	
SCC register area*[2]	—	—	—	—

Note 1. Only for channel 0.

Note 2. Refer to **Section 40.7, SCC Register Descriptions**.

40.2 Register Description

40.2.1 Command Type Register (SD_CMD)

The command type register (SD_CMD) is used to select the command type and response type. The command sequence is started by writing to SD_CMD.

For details on the SD_CMD setting, refer to **Section 40.4.14, Example of SD_CMD Register Setting**.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	MD7	00	R/W	Multiple Block Transfer Mode (enabled at multiple block transfer) 00: CMD12 is automatically issued at multiple block transfer. 01: CMD12 is not automatically issued at multiple block transfer. 10: Setting prohibited 11: Setting prohibited
14	MD6			
13	MD5	0	R/W	Single/Multiple Block Transfer (enabled when the command with data is handled) 0: Single block transfer 1: Multi block transfer
12	MD4	0	R/W	Write/Read Mode (enabled when the command with data is handled) 0: Write (SD/MMC host interfaces -> SD card) 1: Read (SD/MMC host interfaces <- SD card)
11	MD3	0	R/W	Data Mode (Command Type) 0: Command without data transfer (bc, bcr, ac) 1: Command with data transfer (adtc)
10	MD2	000	R/W	Mode/Response Type 000: Normal mode The response type and the transfer mode are selected by SD_CMD[7:0], and the SD_CMD[15:11] setting is disabled. 001: Setting prohibited 010: Setting prohibited 011: Extended mode/No response 100: Extended mode/R1, R5, R6, or R7 response from the SD card 101: Extended mode/R1b response from the SD card 110: Extended mode/R2 response from the SD card 111: Extended mode/R3 or R4 response from the SD card Some commands cannot be used in normal mode. For details, see **Section 40.4.14, Example of SD_CMD Register Setting** to select mode/response type.
9	MD1			
8	MD0			
7	C1	00	R/W	00: CMD 01: ACMD 10: Setting prohibited 11: Setting prohibited
6	C0			
5	CF45	000000	R/W	Command Index These bits specify Command Format[45:40] (command index). [Examples] CMD6: SD_CMD[7:0] = 0000 0110b CMD18: SD_CMD[7:0] = 0001 0010b ACMD13: SD_CMD[7:0] = 0100 1101b
4	CF44			
3	CF43			
2	CF42			
1	CF41			
0	CF40			

Note: SD_CMD cannot be written to when the CBSY bit in SD_INFO2 is 1.

40.2.2 Command Argument Register (SD_ARG)

Command arguments for SD cards are set in the SD command argument registers (SD_ARG). Set the command arguments before writing to SD_CMD.

Note that the argument of CMD12 within command sequences is H'0000 0000 regardless of the setting of SD_ARG.

SD_ARG

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	CF39 to CF8	All 0	R/W	Set command format[39:8] (argument).

SD_ARG1 (Mirror of SD_ARG[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	CF39 to CF24	All 0	R/W	Set command format[39:24] (argument).

40.2.3 Data Stop Register (SD_STOP)

The data stop register (SD_STOP) is used to enable or disable block counting at multiple block transfer, and to control the issuing of CMD12 within command sequences.

Bit	Bit Name	Initial Value	R/W	Description
63 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
18	—	0*1	R/W	Reserved The write value should always be 0.
17	HPIMODE	0*1	R/W	HPI Mode Enable 0: Disables HPI mode. 1: Enables HPI mode.
16	HPICMD	0*1	R/W*3	HPI Command Issue When HPICMD is set to 1 while HPIMODE is 1, the HPI command (CMD12) is issued. This bit is cleared to 0 when reception of the response to CMD12 is completed. The timing with which this bit is set to 1 is as follows. • After reception of the response to CMD12 that was issued by setting the STP bit to 1 has been completed during the CMD6/CMD38 or CMD25 sequence. • After reception of the response to CMD24/CMD25 has been completed After HPICMD is set to 1, do not write 0 to this bit while the CBSY bit in SD_INFO2 is 1. Do not set this bit to 1 when the CBSY bit in SD_INFO2 is 0.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	SEC	0*1	R/W	Block Count Enable*2 0: Disables SD_SECCNT setting value. 1: Enables SD_SECCNT setting value. Set SEC to 1 at multiple block transfer. When SD_CMD is set as follows to start the command sequence while SEC is set to 1, CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT. 1. CMD18 or CMD25 in normal mode (SD_CMD[10:8] = 000) 2. SD_CMD[15:13] = 001 in extended mode (CMD12 is automatically issued, multiple block transfer) When the command sequence is halted because of a communications error or timeout, CMD12 is not automatically issued.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

Bit	Bit Name	Initial Value	R/W	Description
0	STP	0*[1]	R/W	Stop ● When STP is set to 1 during multiple block transfer, CMD12 is issued to halt the transfer through the SD/MMC host interface. However, if a command sequence is halted because of a communications error or timeout, CMD12 is not issued. Although continued buffer access is possible even after STP has been set to 1, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. ● When STP has been set to 1 during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD12 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. ● When STP has been set to 1 during transfer for single block read, the access end flag is set immediately after setting of the STP bit and CMD12 is not issued. ● When STP is set to 1 during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. ● When STP is set to 1 after a command sequence has been completed, CMD12 is not issued and the access end flag is not set. ● Set STP to 1 after the response end flag has been set. ● Set STP to 0 after the response end flag has been set.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

Note 3. It is effective only if 1 is written.

40.2.4 Block Count Register (SD_SECCNT)

The block count register (SD_SECCNT) is used to specify the number of transfer blocks at multiple block transfer.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	CNT31 to CNT0	All 0	R/W	Number of Transfer Blocks*[1] When H’0000 0001 is set, the number of transfer blocks is 1. : When H’0000 FFFF is set, the number of transfer blocks is 65535. : When H’FFFF FFFF is set, the number of transfer blocks is 4294967295. Do not set this register to H’0000 0000 if multiple blocks are to be transferred.

Note 1. Do not change the value of these bits when the CBSY bit in SD_INFO2 is set to 1.

40.2.5 SD Card Response Registers (SD_RSP)

The SD card response registers (SD_RSP) hold the response from the SD card.

SD_RSP10

Bit	Bit Name	Initial Value	R/W	Description
63 to 0	R71 to R8	All 0	R	Hold the response from the SD card

SD_RSP1 (Mirror of SD_RSP10[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R39 to R24	All 0	R	Hold the response from the SD card

SD_RSP32 (Mirror of SD_RSP10[63:32])

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 0	R71 to R40	All 0	R	Hold the response from the SD card

SD_RSP3 (Mirror of SD_RSP32[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R71 to R56	All 0	R	Hold the response from the SD card

SD_RSP54

Bit	Bit Name	Initial Value	R/W	Description
63 to 56	—	All 0	R	These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
55 to 0	R127 to R72	All 0	R	Hold the response from the SD card

SD_RSP5 (Mirror of SD_RSP54[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	R103 to R88	All 0	R	Hold the response from the SD card

SD_RSP76 (Mirror of SD_RSP54[63:32])

Bit	Bit Name	Initial Value	R/W	Description
63 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 0	R127 to R104	All 0	R	Hold the response from the SD card

SD_RSP7 (Mirror of SD_RSP76[31:16])

Bit	Bit Name	Initial Value	R/W	Description
63 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	R127 to R120	All 0	R	Hold the response from the SD card

Table 40.3 lists the response types and corresponding SD_RSP registers.

Table 40.3 Response Types and Corresponding SD_RSP Registers

Response Types	SD_RSP Registers
R1, R1b[39:8]	SD_RSP10 SD_RSP54*[1]
R2[127:8]	SD_RSP54 and SD_RSP10
R3[39:8]	SD_RSP10
R4[39:8]	SD_RSP10
R5[39:8]	SD_RSP10
R6[39:8]	SD_RSP10
R7[39:8]	SD_RSP10

Note 1. The response to CMD18 and to CMD25 is stored in both R[39:8] and R[103:72].
This makes it possible to confirm the response to CMD18 and CMD25 by reading R[103:72] even if the response to automatic CMD12 is stored in R[39:8].

40.2.6 SD Card Interrupt Flag Register (SD_INFO1)

The SD card interrupt flag register 1 (SD_INFO1) indicates the response end and access end in the command sequence. This register also indicates the card detect/write protect state.

For CMD12 and CMD52 (SDIO abort) at multiple block transfer, INFO0 is not set but only INFO2 is set.

Even if the command sequence is halted because of a communications error or timeout, INFO0 or INFO2 is set.

INFO10, INFO9, and INFO8 change depending on the SDDAT3 state after a reset is released and continue to change in 4-bit transfer mode.

To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	HPIRES	0*2	R/W*1	Response Reception Completion [Setting condition] When reception of the response to CMD12 that was issued by setting the STP bit to 1 is completed during the CMD6/CMD38 or CMD25 sequence in HPI mode. [Clearing condition] When 0 is written to HPIRES
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
10	INFO10	Unknown	R	Indicates the SDDAT3 state. 0: SDDAT3 is set to 0. 1: SDDAT3 is set to 1.
9	INFO9	0	R/W*1	SDDAT3 Card Insertion [Setting condition] After change in SDDAT3 from 0 to 1, two cycles of 2 P1ϕ have elapsed with SDDAT3 held 1. [Clearing condition] When 0 is written to INFO9
8	INFO8	0	R/W*1	SDDAT3 Card Removal [Setting condition] After change in SDDAT3 from 1 to 0, two cycles of 2 P1ϕ have elapsed with SDDAT3 held 0. [Clearing condition] When 0 is written to INFO8
7	INFO7	Unknown	R	Write Protect Indicates the ISDWP state. 0: ISDWP is set to 1. 1: ISDWP is set to 0.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
5	INFO5	Unknown	R	Indicates the ISDCD state. 0: Indicates that Mcycle has elapsed with ISDCD held 1. 1: Indicates that Mcycle has elapsed with ISDCD held 0. Mcycle is set by bits 3 to 0 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
4	INFO4	0	R/W*1	ISDCD Card Insertion [Setting condition] After change in ISDCD from 1 to 0, Mcycle has elapsed with ISDCD held 0. [Clearing condition] When 0 is written to INFO4 Mcycle is set by bits 3 to 0 in SD_OPTION.
3	INFO3	0	R/W*1	ISDCD Card Removal [Setting condition] After change in ISDCD from 0 to 1, Mcycle has elapsed with ISDCD held 1. [Clearing condition] When 0 is written to INFO3 Mcycle is set by bits 3 to 0 in SD_OPTION.
2	INFO2	0*2	R/W*1	Access End [Setting conditions] 1. When read access to the buffer is completed in the case of transfer for single block read 2. When read access to the buffer for the last block of data is completed in the case of transfer for multiple block read 3. When read access to the buffer and reception of the response to CMD12 are completed in the case of transfer for multiple block read with automatic issuing of CMD12 4. When reception of the busy state after reception of the CRC status is completed in the case of transfer for single block write 5. When reception of the busy state after reception of the CRC status of the last block of data is completed in the case of transfer for multiple block write 6. When reception of the response busy state for CMD12 is completed in the case of transfer for multiple block write with automatic issuing of CMD12 7. When reception of the response to CMD12 that was issued by setting the STP bit to1 is completed in the case of transfer for multiple block read 8. When reception of the response busy state for CMD12 that was issued by setting the STP bit to1 is completed in the case of transfer for multiple block write 9. When reception of the response to CMD52 that was issued by setting the IOABT bit to1 is completed in the case of transfer for multiple block read 10. When reception of the response to CMD52 that was issued by setting the IOABT bit to1 is completed in the case of transfer for multiple block write In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO2 When the access end bit is set to 1, the command sequence is terminated.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.

Bit	Bit Name	Initial Value	R/W	Description
0	INFO0	0*2	R/W*1	Response End [Setting conditions] 1. When the reception of the response is completed 2. When the transmission of the command not requiring a response is completed, 3. When receiving busy reception after R1b response 4. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block read 5. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block write, In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO0 When issuing a command without data, the command sequence ends when the response end is set to 1.

Note 1. It is effective only if 0 is written.

Note 2. The value is initialized by a reset and also in the case of a reset by the SDRST bit in SOFT_RST.

40.2.7 SD Card Interrupt Flag Register (SD_INFO2)

The SD card interrupt flag register 2 (SD_INFO2) indicates the access status of the SD buffer (SD_BUF) and SD card. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	ILA	0*2	R/W*1	Illegal Access Error [Setting conditions] 1. Writing of data to SD_CMD within a command sequence (CBSY=1) 2. When SD_CMD[11] = 1 (command with data transfer) and SD_CMD[7:0] = 0000 1100b (CMD12) are set in SD_CMD [Clearing condition] When 0 is written to ILA
14	CBSY	0*2	R	Command Type Register Busy 0: A command sequence has been completed. 1: A command sequence is being executed.
13	SCLKDIVEN	1*2	R	0: The SD bus (CMD, DAT) is busy. Do not attempt to write to the SD_CLK_CTRL register. 1: The SD bus (CMD, DAT) is not busy. When a command sequence is started by writing to SD_CMD, the CBSY bit is set to 1 and, at the same time, the SCLKDIVEN bit is set to 0. The SCLKDIVEN bit is set to 1 after 8 cycles of SDCLK have elapsed after setting of the CBSY bit to 0 due to completion of the command sequence.
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
11	—	0	R/W	Reserved The write value should always be 1.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
9	BWE	0*2	R/W*1	SD_BUF Write Enable 0: Data cannot be written in SD_BUF0. 1: Data can be written in SD_BUF0. [Setting conditions] 1. When SD_BUF is empty at single block transfer 2. When either bank 1 or bank 2 of SD_BUF is empty at multiple block transfer [Clearing conditions] 1. When 0 is written to BWE 2. Writing of a block of data to SD_BUF by DMA transfer When data is written to SD_BUF0 by the CPU, clear BWE and then write amount of data specified by SD_SIZE*3

Bit	Bit Name	Initial Value	R/W	Description
8	BRE	0*2	R/W*1	SD_BUF Read Enable 0: Data cannot be read from SD_BUF0. 1: Data can be read from SD_BUF0. [Setting conditions] 1. When data set in SD_SIZE is stored in SD_BUF at single block transfer 2. When data set in SD_SIZE is stored in either bank 1 or bank 2 of SD_BUF at multiple block transfer [Clearing conditions] 1. When 0 is written to BRE 2. Reading of a block of data from SD_BUF by DMA transfer When data is read from SD_BUF0 by the CPU, clear BRE and then read amount of data specified by SD_SIZE*3. Even if a CRC error or an END error occurs while block data is read, data is stored in SD_BUF and BRE is set.
7	DAT0	Unknown	R	SDDAT0 Indicates the SDDAT0 state. 0: SDDAT0 is set to 0. 1: SDDAT0 is set to 1. If the data timeout (ERR3) is set but the response timeout (ERR6) is not set after the Erase command has been issued, the end of the Erase sequence (DAT0 = 1) is confirmed by polling DAT0. If a communications error or timeout occurs during a write sequence, the DAT0 bit may retain the value 0. While the SD clock (SDCLK) is stopped, the DAT0 bit retains the value before the clock is stopped.
6	ERR6	0*2	R/W*1	Response Timeout [Setting condition] When a response is not received even after 640 cycles of SDCLK have elapsed (including a response to a command issued within a command sequence*5) [Clearing condition] When 0 is written to ERR6 The command sequence is halted by a response timeout.*4
5	ERR5	0*2	R/W*1	SD_BUF Illegal Read Access [Setting conditions] 1. When SD_BUF is empty while SD_BUF0 is read 2. When data with a CRC error or END error is read from SD_BUF0 [Clearing condition] When 0 is written to ERR5
4	ERR4	0*2	R/W*1	SD_BUF Illegal Write Access [Setting conditions] 1. When data is written to SD_BUF0 while it is not in the data read/write command state 2. When data is written to SD_BUF0 while SD_BUF is full 3. When data is written to SD_BUF0 while an error occurs in the CRC status or CRC status length 4. When data is written to SD_BUF0 while the interface remains in a busy state for at least Ncycle after the CRC status [Clearing condition] When 0 is written to ERR4 Ncycle is set by bits 7 to 4 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
3	ERR3	0*[2]	R/W*[1]	Data Timeout (except response timeout) [Setting conditions] 1. After R1b response, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 2. After CRC status, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 3. After write data, the CRC status is not received even after Ncycle has elapsed. 4. After read command, read data is not received even after Ncycle has elapsed. 5. After CMD12 has been issued within a command sequence, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 6. After the reception of read data, read data for the next block are not received even after Ncycle has elapsed. 7. After release of the read wait state, read data for the next block are not received even after Ncycle has elapsed. [Clearing condition] When 0 is written to ERR3 Ncycle is set by bits 7 to 4 in SD_OPTION. The command sequence is halted by the data timeout.
2	ERR2	0*[2]	R/W*[1]	END Error [Setting conditions] 1. When an error occurs in the response length (and the end bit has not been detected) 2. When an error occurs in the read data length (and the end bit has not been detected among the valid bits) 3. When an error occurs in the CRC status length (and the end bit has not been detected) 4. An error in the length of a response to a command issued within a command sequence*[5] (i.e. the end bit has not been detected) [Clearing condition] When 0 is written to ERR2 The command sequence is halted by the End error.*[4]
1	ERR1	0*[2]	R/W*[1]	CRC Error [Setting conditions] 1. When an error occurs in the CRC status (i.e. the received CRC status was not 010') 2. When a CRC error occurs in the read data 3. When a CRC error occurs in the response 4. A CRC error in the response to a command issued within a command sequence*[5] [Clearing condition] When 0 is written to ERR1 The command sequence is halted by the CRC error.*[4]
0	ERR0	0*[2]	R/W*[1]	CMD Error [Setting conditions] 1. The command index of the transmitted command differing from the command index of the received response 2. The command index of a command issued within a command sequence*[5] differing from the command index of the received response [Clearing condition] When 0 is written to ERR0 The command sequence is halted by the CMD error.*[4]

Note 1. It is effective only if 0 is written.

Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 3. When the WMODE bit in HOST_MODE is 0, the single byte from the fraction of a full 16-bit unit is regarded as excess data due to an odd value for the number of bytes setting in SD_SIZE. When the WMODE bit in HOST_MODE is 1, the single byte or three bytes from the fraction of a full 64-bit unit are regarded as excess data due to an odd value for the number of bytes setting in SD_SIZE, or the two bytes from the fraction of a full 64-bit unit are regarded as excess data due if the value for the number of bytes setting in SD_SIZE is even but is not on a four-byte boundary.

Note 4. After the C52PUB bit in SDIO_MODE has been set to 1, if a communications error or timeout for response occurs in response to the CMD52 that is issued, since the command sequence has not been completed, complete the sequence with error processing as in usage examples in **Figure 40.16** under **Section 40.4.8, IO_RW_EXTENDED (CMD53/Multiple Block Read)** or in **Figure 40.19** under **Section 40.4.9, IO_RW_EXTENDED (CMD53/Multiple Block Write)**.

Note 5. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

40.2.8 SD_INFO1 Interrupt Mask Register (SD_INFO1_MASK)

The SD_INFO1 interrupt mask register (SD_INFO1_MASK) is used to enable or disable the SD_INFO1 interrupt.

When 0 is set in SD_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	IMASK16	1	R/W	HPIRES interrupt masked
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	IMASK9	1	R/W	INFO9 interrupt masked
8	IMASK8	1	R/W	INFO8 interrupt masked
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
4	IMASK4	1	R/W	INFO4 interrupt masked
3	IMASK3	1	R/W	INFO3 interrupt masked
2	IMASK2	1	R/W	INFO2 interrupt masked
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	IMASK0	1	R/W	INFO0 interrupt masked

40.2.9 SD_INFO2 Interrupt Mask Register (SD_INFO2_MASK)

The SD_INFO2 interrupt mask register (SD_INFO2_MASK) is used to enable or disable the SD_INFO2 interrupt.

When 0 is set in SD_INFO2_MASK while the corresponding flag in SD_INFO2 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	IMASK	1	R/W	ILA interrupt masked
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11	—	1	R/W	Reserved The write value should always be 1.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
9	BMASK1	1	R/W	BWE interrupt masked
8	BMASK0	1	R/W	BRE interrupt masked
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
6	EMASK6	1	R/W	ERR6 interrupt masked
5	EMASK5	1	R/W	ERR5 interrupt masked
4	EMASK4	1	R/W	ERR4 interrupt masked
3	EMASK3	1	R/W	ERR3 interrupt masked
2	EMASK2	1	R/W	ERR2 interrupt masked
1	EMASK1	1	R/W	ERR1 interrupt masked
0	EMASK0	1	R/W	ERR0 interrupt masked

40.2.10 SD Clock Control Register (SD_CLK_CTRL)

The SD clock control register (SD_CLK_CTRL) is used to control the SD clock (SDCLK) output and to set the frequency. Set SCLKEN to 1 before writing to SD_CMD to issue a command. Do not write to SD_CLK_CTRL while the SCLKDIVEN bit in SD_INFO2 is set to 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	—	0	R/W	Reserved The write value should always be 0.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
10	—	0[*1]	R/W	Reserved The write value should always be 0.
9	SDCLKOFFEN	0	R/W	SD Clock (SDCLK) Output Automatic Control Enable 0: Automatic control for SD clock (SDCLK) output is disabled. 1: Automatic control for SD clock (SDCLK) output is enabled. This function of automatic control for SD clock (SDCLK) output causes SDCLK output only within a command sequence. The timing with which SDCLK output starts and stops is as follows. SDCLK output starts after writing to SD_CMD. SDCLK output stops when 8 cycles of SDCLK have elapsed after the end of the command sequence. In addition, SDCLK is fixed to 0 while SCLKEN of SD_CLK_CTRL is 0, regardless of the value of this bit.
8	SCLKEN	0[*1]	R/W[*2]	SD Clock (SDCLK) Output Control Enable 0: SD clock (SDCLK) output is disabled. The SDCLK signal is fixed 0. 1: SD clock (SDCLK) output is enabled.
7	DIV7	0	R/W[*2]	SD Clock (SDCLK) 1000 0000b: (SDxϕ/4)/512 0100 0000b: (SDxϕ/4)/256 0010 0000b: (SDxϕ/4)/128 0001 0000b: (SDxϕ/4)/64 0000 1000b: (SDxϕ/4)/32 0000 0100b: (SDxϕ/4)/16 0000 0010b: (SDxϕ/4)/8 0000 0001b: (SDxϕ/4)/4 0000 0000b: (SDxϕ/4)/2 1111 1111b: SDxϕ/4 (x = 0, 1) Other settings are prohibited.
6	DIV6	0	R/W[*2]	
5	DIV5	1	R/W[*2]	
4	DIV4	0	R/W[*2]	
3	DIV3	0	R/W[*2]	
2	DIV2	0	R/W[*2]	
1	DIV1	0	R/W[*2]	
0	DIV0	0	R/W[*2]	

Note 1. This initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. Writing to SD_CLK_CTRL is impossible when the CBSY bit in SD_INFO2 is 1.

Notes on when setting the SD clock (SDCLK) to P1ϕ (DIV[7:0] = 1111 1111b)

When changing the setting of bits DIV[7:0] to 1111 1111b, or from 1111 1111b to other setting, perform the following processing before writing to SD_CMD.

(1) Set the SCLKEN bit to 0 by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)

(2) Change the setting of bits DIV[7:0] by writing to SD_CLK_CTRL. (Do not change the setting of bits other than DIV[7:0] at this time. The SCLKEN bit should retain the value 0.)

(3) Set the SCLKEN bit to 1 by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)

Also when changing the setting of bits DIV[7:0] to 1111 1111b after having set the SDRST bit in SOFT_RST to 0 and then changed it to 1, perform this processing before writing to SD_CMD.

40.2.11 Transfer Data Length Register (SD_SIZE)

The transfer data length register (SD_SIZE) is used to specify the transfer data size.

Bit	Bit Name	Initial Value	R/W	Description
63 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11, 10	—	All 0	R	Reserved
9 to 0	LEN9 to LEN0	1000000000	R/W	Transfer Data Size*[1] These bits specify a size between 1 and 512 bytes for single block transfer. In cases of multiple block transfer with automatic issuing of CMD12 (CMD18 and CMD25), the only specifiable transfer data size is 512 bytes. Furthermore, in cases of multiple block transfer without automatic issuing of CMD12, as well as 512 bytes, 32, 64, 128, and 256 bytes are specifiable. However, in the reading of 32, 64, 128, and 256 bytes for the transfer of multiple blocks, this is restricted to multiple block transfer by CMD53. Additionally, if a command accompanies data transfer, do not set these bits to 0. Do not specify a data size larger than 512 bytes.

Note 1. Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1.

40.2.12 SD Card Access Control Option Register (SD_OPTION)

The SD card access control option register (SD_OPTION) is used to set the bus width and timeout counter.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	WIDTH	0*1	R/W	Bus width*2 {WIDTH,WIDTH8} = 01: 8 bits width {WIDTH,WIDTH8} = 00: 4 bits width {WIDTH,WIDTH8} = 10,11: 1 bit width In the case of writing of one-byte block, 8 bits width cannot be specified for the bus width. Change the bus width to 4 bits or 1 bit before writing one-byte block.
14	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
13	WIDTH8	0*1	R/W	Bus width*2 See the description of the WIDTH bit.
12 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	EXTOP	0*1	R/W	Timeout Mode Select 0: Bits TOP27 to TOP24 specify the timeout count from SDCLK*2^{13} to SDCLK*2^{27}. 1: Bits TOP27 to TOP24 specify the timeout count from SDCLK*2^{14} to SDCLK*2^{28}.
8	TOUTMASK	0*1	R/W	Timeout Mask 0: Enables timeout. 1: Disables timeout. (The ERR6 and ERR3 bits in SD_INFO2 and the E6 to E0 bits in SD_ERR_STS2 are not set.) If a timeout occurs while it is disabled, perform a software reset to terminate a command sequence.
7	TOP27	1*1	R/W	Timeout Counter*2 0000: SDCLK*2^{13} 0001: SDCLK*2^{14} : 1101: SDCLK*2^{26} 1110: SDCLK*2^{27} 1111: Setting prohibited
6	TOP26	1*1	R/W	
5	TOP25	1*1	R/W	
4	TOP24	0*1	R/W	
3	CTOP24	1*1	R/W	Card Detect Time Counter 0000: SDCLK*2^{10} 0001: SDCLK*2^{11} : 1101: SDCLK*2^{23} 1110: SDCLK*2^{24} 1111: Setting prohibited
2	CTOP23	1*1	R/W	
1	CTOP22	1*1	R/W	
0	CTOP21	0*1	R/W	

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1.

40.2.13 SD Error Status Register 1 (SD_ERR_STS1)

The SD error status register 1 (SD_ERR_STS1) indicates the CRC status, CRC error, End error, and CMD error.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 15	—	Unknown	R	Reserved These bits are always read as unknown. The write value should always be 0.
14	E14	0*[1]	R	These bits hold the CRC status. (normal: 010)
13	E13	1*[1]		
12	E12	0*[1]		
11	E11	0*[1]	R	Set to 1 when an error occurs in the CRC status.
10	E10	0*[1]	R	Set to 1 when a CRC error occurs in the read data.
9	E9	0*[1]	R	Set to 1 when a CRC error occurs in the response to a command issued within a command sequence*[2]. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E8.
8	E8	0*[1]	R	Set to 1 when a CRC error occurs in a response (other than a response to a command issued within a command sequence*[2]).
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5	E5	0*[1]	R	Set to 1 when an error occurs in the CRC status length (and the end bit has not been detected).
4	E4	0*[1]	R	Set to 1 when an error occurs in the read data length (and the end bit has not been detected among the valid bits).
3	E3	0*[1]	R	Set to 1 when an error occurs in the response length to a command issued within a command sequence*[2]. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0*[1]	R	Set to 1 when an error occurs in the response length (other than a response to a command issued within a command sequence*[2]).
1	E1	0*[1]	R	Set to 1 when an error occurs in the command index of the response to a command issued within a command sequence*[2]. In cases where CMD12 is issued by setting a command index in SD_CMD, this is Indicated in E0.
0	E0	0*[1]	R	Set to 1 when an error occurs in the command index of a response (other than a response to a command issued within a command sequence*[2]).

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

40.2.14 SD Error Status Register 2 (SD_ERR_STS2)

The SD error status register 2 (SD_ERR_STS2) indicates the timeout state. Ncycle is set by bits 7 to 4 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
63 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
6	E6	0*[1]	R	Set to 1 when the interface remains in a busy state for at least Ncycle after the CRC status.
5	E5	0*[1]	R	Set to 1 when the CRC status is not received even after Ncycle has elapsed after data writing.
4	E4	0*[1]	R	Set to 1 when read data is not received even after Ncycle has elapsed after the command has been read. Set to 1 when read data for the next block are not received even after Ncycle has elapsed after the reception of read data. Set to 1 when read data for the next block are not received even after Ncycle has elapsed after release of the read wait state.
3	E3	0*[1]	R	Set to 1 when the interface remains in a busy state for at least Ncycle after CMD12 has been issued within a command sequence. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0*[1]	R	Set to 1 when the interface remains in a busy state for at least Ncycle after R1b response.
1	E1	0*[1]	R	Set to 1 when the response to a command issued within a command sequence*[2] is not received even after 640 cycles of SDCLK have elapsed. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E0.
0	E0	0*[1]	R	Set to 1 when the response (other than a response to a command issued within a command sequence*[2]) is not received even after 640 cycles of SDCLK have elapsed.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1.

40.2.15 SD Buffer Read/Write Register (SD_BUF0)

Bit	Bit Name	Initial Value	R/W	Description
63 to 0	BUF63 to BUF0	Unknown	R/W	When writing to the SD card, the write data is written to this register. When reading from the SD card, the read data is read from this register. This register is internally connected to two 512-byte buffers (SD_BUF). When both buffers are not empty at multiple block read, suspend data reception by stopping the SD clock. When either buffer becomes empty, restart data reception by starting supply of the SD clock.

Note: When using the DMAC, the bus width should be fixed at 64 bits.

40.2.16 SDIO Mode Control Register (SDIO_MODE)

The SDIO mode control register (SDIO_MODE) controls the CMD52 issuance and the read wait state at multiple block transfer, and the reception of SDIO interrupt. C52PUB and IOABT should not be set to 1 simultaneously.

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	C52PUB	0	R/W	SDIO None Abort ● When C52PUB is set to 1 in the CMD53 (multiple block) write sequence, CMD52 is automatically issued between blocks if SD_BUF becomes empty.C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1 while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1. ● When C52PUB and RWREQ are set to 1 in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks and CMD52 is automatically issued. C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1 while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1. ● If C52PUB is set to 1 in the CMD53 (multiple block) read sequence, be sure to set RWREQ to 1 as well as C52PUB. ● Set SD_ARG before setting C52PUB to 1. ● Set C52PUB to 1 after the response end flag has been set.
8	IOABT	0	R/W	SDIO Abort ● When IOABT is set to 1 in the CMD53 (multiple block) sequence, the CMD53 sequence is halted and CMD52 is issued. However, if a command sequence is halted because of a communications error or timeout, CMD52 is not issued. Although continued buffer access is possible even after IOABT has been set to 1, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. Set SD_ARG before setting IOABT to 1. ● When IOABT has been set to 1 during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD52 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. ● When IOABT has been set to 1 during transfer for single block read, the access end flag is set immediately after setting of IOABT and CMD52 is not issued. ● When IOABT is set to 1 during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. ● When IOABT is set to 1 after a command sequence has been completed, CMD52 is not issued and the access end flag is not set. ● Set IOABT to 1 after the response end flag has been set. Set IOABT to 0 after the access end flag has been set.
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

Bit	Bit Name	Initial Value	R/W	Description
2	RWREQ	0	R/W	Read Wait Request When RWREQ is set to 1 in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks. [Read wait state releasing] (1) The read wait state is released, when RWREQ is cleared to 0 in the read wait state. (2) When IOABT is set to 1 in the read wait state, RWREQ is automatically cleared to 0 after CMD52 has been issued, and then the read wait state is released. (3) When C52PUB and RWREQ are set to 1 simultaneously in the CMD53 (multiple block) read sequence, the read wait state is not automatically released. Therefore, after the CMD52 response is received, clear RWREQ. (Be sure to set RWREQ and C52PUB simultaneously.) When RWREQ is set to 1 while the last block in the CMD53 (multiple block) read sequence is transferred, the read wait state is not entered and RWREQ is automatically cleared to 0 by setting access end. Set RWREQ to 1 after the response end flag has been set.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.
0	IOMOD	0	R/W	SDIO Mode*[1] 0: Disables the SD/MMC host interfaces to receive SDIO interrupt from the SDIO card 1: Enables the SD/MMC host interfaces to receive SDIO interrupt from the SDIO card

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

40.2.17 SDIO Interrupt Flag Register (SDIO_INFO1)

The SDIO interrupt flag register (SDIO_INFO1) indicates the status regarding to the SDIO card access. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	EXWT	0*2	R/W*1	[Setting condition] While the last block in the CMD53 (multiple block) read sequence is transferred, RWREQ in SDIO_MODE is set to 1. [Clearing condition] When 0 is written to EXWT
14	EXPUB52	0*2	R/W*1	[Setting conditions] 1. While the last block in the CMD53 (multiple block) sequence is transferred, C52PUB in SDIO_MODE is set to 1. 2. While C52PUB is set to 1 in the CMD53 (multiple block) write sequence, the last block is transferred. [Clearing condition] When 0 is written to EXPUB52
13 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2, 1	—	All 0*2	R/W	Reserved The write value should always be 1. The value may change during operation.
0	IOIRQ	0*2	R/W*1	[Setting condition] When SDIO interrupt from an SDIO card is received while IOMOD in SDIO_MODE is set to 1. [Clearing condition] When 0 is written to IOIRQ*3

Note 1. It is effective only if 0 is written.

Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 3. Before clearing this bit, access the SDIO card to negate the SDIO interrupt signal from the SDIO card. If the interrupt signal is not negated, this bit may be set again.

40.2.18 SDIO_INFO1 Interrupt Mask Register (SDIO_INFO1_MASK)

The SDIO_INFO1 interrupt mask register (SDIO_INFO1_MASK) enables or disables the SD_INFO1 interrupt. When 0 is set in SDIO_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15	MEXWT	1	R/W	EXWT interrupt masked
14	MEXPUB52	1	R/W	EXPUB52 interrupt masked
13 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2	—	1	R/W	Reserved The write value should always be 1.
1	—	1	R/W	Reserved The write value should always be 1.
0	IOMSK	1	R/W	IOIRQ interrupt masked

40.2.19 DMA Mode Enable Register (CC_EXT_MODE)

The DMA mode enable register (CC_EXT_MODE) enables the DMA transfer.

Bit	Bit Name	Initial Value	R/W	Description
63 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9, 8	—	All 0	R/W	Reserved The write value should always be 0.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5	—	0	R/W	Reserved The write value should always be 0.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to this bit.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	DMASDRW	0	R/W	SD_BUF Read/Write DMA Transfer*[1] 0: The SD_BUF read/write DMA transfer is disabled. 1: The SD_BUF read/write DMA transfer is enabled.
0	—	0	R/W	Reserved The write value should always be 0.

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

40.2.20 Software Reset Register (SOFT_RST)

The software reset register (SOFT_RST) sets a software reset. Also use this register to check that release from the reset state has been completed before attempting to use the SD/MMC host interfaces and before attempting access to the other registers.

Bit	Bit Name	Initial Value	R/W	Description
63 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2, 1	—	11	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SDRST	1	R/W	Software Reset of SD Interface Unit 0: Reset 1: Reset released

40.2.21 Version Register (VERSION)

The version register (VERSION) indicates the version of the SD/MMC host interfaces.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15, 14	UR7, UR6	11	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
13, 12	UR5, UR4	00	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11 to 8	UR3 to UR0	H'C	R	Version of Renesas' IP
7 to 0	IP7 to IP0	H'10	R	Version of introductory IP

40.2.22 Host Interface Mode Setting Register (HOST_MODE)

The host interface mode setting register (HOST_MODE) selects the width for access to the data bus.

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	BUSWIDTH	0	R/W	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0: 16-bit access 1: 32-bit access This bit is enabled while the WMODE bit is set to 1.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	ENDIAN	0	R/W	SD_BUF0 data swap
0	WMODE	0	R/W	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0: 64-bit access 1: 16-bit or 32-bit access

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

Note 2. When using the built-in DMAC of this module, fix the bus width to 64 bits.

40.2.23 SD Interface Mode Setting Register (SDIF_MODE)

The SD interface mode setting register specifies transfer mode.

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
9	—	0*[1]	R/W	Reserved The write value should always be 0.
8	NOCHKCR	0*[1]	R/W	CRC Check Mask (test command for MMC supported) Enables or disables checking of the CRC16 and CRC status. 0: Enables the CRC check. 1: Disables the CRC check (the CRC16 value is ignored at read, and the CRC status is not detected at write)
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

40.2.24 SD Status Register (SD_STATUS)

The effective bit of the SD status register controls the output value on the SD0_RST# pin.

Bit	Bit Name	Initial Value	R/W	Description
63 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	SD_RST	0	R/W	Controls the output value on the SD0_RST# pin. 0: The output value on the SD0_RST# pin is 0. 1: The output value on the SD0_RST# pin is 1.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.

40.2.25 DMAC Transfer Mode Register (DM_CM_DTRAN_MODE)

The DMAC Transfer Mode Register (DM_CM_DTRAN_MODE) sets the operation mode of the module built-in DMAC.

High 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
17, 16	CH_NUM	00	R/W	DMAC channel selector 00 : SD down stream 01 : SD up stream Other settings are prohibited.
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
5, 4	BUS_WIDTH [1:0]	11	R/W	Bus width selector 11: 64-bit Other settings are prohibited.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

40.2.26 DMAC Transfer Control Register (DM_CM_DTRAN_CTRL)

The DMAC Transfer Control Register (DM_CM_DTRAN_CTRL) controls the module built-in DMAC operation. High 32 bits (bit 63-32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8	—	0	R/W	Reserved The write value should always be 0.
7 to 1	—	All 0	R	Reserved
0	DM_START	0	R/W	DMAC Start Writing 1 to this bit starts DMAC operation. This bit is automatically cleared when DMA transfer is started.

40.2.27 DMAC Reset Register (DM_CM_RST)

High 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 10	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
9	DTRANRST1	1	R/W	Soft resets of the module built-in DMAC channel 1 0: Reset 1: Reset released
8	DTRANRST0	1	R/W	Soft resets of the module built-in DMAC channel 0 0: Reset 1: Reset released
7 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
0	SEQRST	1	R/W	Soft resets of the sequencer 0: Reset 1: Reset released

NOTE

Make sure there is no communication to the SD/MMC device before applying a software reset using this register.

40.2.28 DMAC Interrupt Register 1 (DM_CM_INFO1)

The DMAC interrupt register 1 (DM_CM_INFO1) indicates the status of the module built-in DMAC and a sequencer. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
20	DTRANEND1	0*[1]	R/W	Module built-in DMAC Channel 1 Transfer End [Setting conditions] 1. When transfer of DMAC channel 1 is completed 2. When an error occurs on DMAC channel 1 [Clearing condition] When 0 is written to DTRANEND1
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
16	DTRANEND0	0*[2]	R/W	Module built-in DMAC Channel 0 Transfer End [Setting conditions] 1. When transfer of DMAC channel 0 is completed 2. When an error occurs on DMAC channel 0 [Clearing condition] When 0 is written to DTRANEND0
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	SEQEND	0*[3]	R/W	Sequencer Operation End [Setting conditions] 1. When operation of a sequencer is completed 2. When a sequencer error occurs [Clearing condition] When 0 is written to SEQEND

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0.

Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0.

Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0.

40.2.29 DM_CM_INFO1 Interrupt Mask Register (DM_CM_INFO1_MASK)

The DM_CM_INFO1 interrupt mask register (DM_CM_INFO1_MASK) enables or disables the DM_CM_INFO1 interrupt. When 0 is set in DM_CM_INFO1_MASK while the corresponding flag in DM_CM_INFO1 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 21	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
20	DTRANEND1_MASK	1	R/W	DTRANEND1 interrupt masked
19 to 17	—	All 1	R/W	Reserved The write value should always be 1.
16	DTRANEND0_MASK	1	R/W	DTRANEND0 interrupt masked
15 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SEQEND_MASK	1	R/W	SEQEND interrupt masked

40.2.30 DMAC Interrupt Register 2 (DM_CM_INFO2)

The DMAC interrupt register 2 (DM_CM_INFO2) indicates the status of the module built-in DMAC and a sequencer. To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Bit	Bit Name	Initial Value	R/W	Description
63 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
19, 18	—	00	R	Reserved The write value should always be 0.
17	DTRANERR1	0*1	R/W	Module built-in DMAC Channel 1 Error [Setting condition] When an error occurs on DMAC channel 1 [Clearing condition] When 0 is written to DTRANERR1
16	DTRANERR0	0*2	R/W	Module built-in DMAC Channel 0 Error [Setting condition] When an error occurs on the DMAC channel 0 [Clearing condition] When 0 is written to DTRANERR0
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	SEQERR	0*3	R/W	Sequencer error [Setting condition] When a sequencer error occurs [Clearing condition] When 0 is written to SEQERR

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0.

Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0.

Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0.

40.2.31 DM_CM_INFO2 Interrupt Mask Register (DM_CM_INFO2_MASK)

The DM_CM_INFO2 interrupt mask register (DM_CM_INFO2_MASK) enables or disables the DM_CM_INFO2 interrupt. When 0 is set in DM_CM_INFO2_MASK while the corresponding flag in DM_CM_INFO2 is set, an interrupt occurs.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 20	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
19, 18	—	11	R	Reserved The write value should always be 1.
17	DTRANERR1_MASK	1	R/W	DTRANERR1 interrupt masked
16	DTRANERR0_MASK	1	R/W	DTRANERR0 interrupt masked
15 to 1	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
0	SEQERR_MASK	1	R/W	SEQERR interrupt masked

40.2.32 DMAC Transfer Address Register (DM_DTRAN_ADDR)

The DMAC Transfer Address Register (DM_DTRAN_ADDR) sets the transfer destination and source address of the module built-in DMAC.

Higher-order 32 bits (bit 63 to 32) are read only 0.

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
31 to 3	DADDR	All 0	R/W	Destination address / Source address (8 byte unit) Note that the value of DM_DTRAN_ADDR + transfer data length is less than or equal to 2^{32}.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.

40.3 Operation

40.3.1 SD Interface

(1) SD Data Format

When data is read from the SD card, the procedure is as follows.

1. The SD/MMC host interface receives data from the SD card via the SDDAT signal. (SDDAT signal: see **Figure 40.2**, **Figure 40.3** and **Figure 40.4**.)
2. The receive data is stored in SD_BUF of the SD/MMC host interfaces. (SD_BUF store data: see **Figure 40.5**)
3. The data stored in SD_BUF is read from SD_BUF0. (Reading from SD_BUF0: see **Table 40.4**)

When data is written to the SD card, the above procedure will be reversed.

When accessing SD_BUF0, caution should be taken for the transfer order in SDDAT and the store order in SD_BUF. In addition, data stored in SD_BUF0 can be replaced in bytes with the EXT_SWAP. (See **Figure 40.5**)

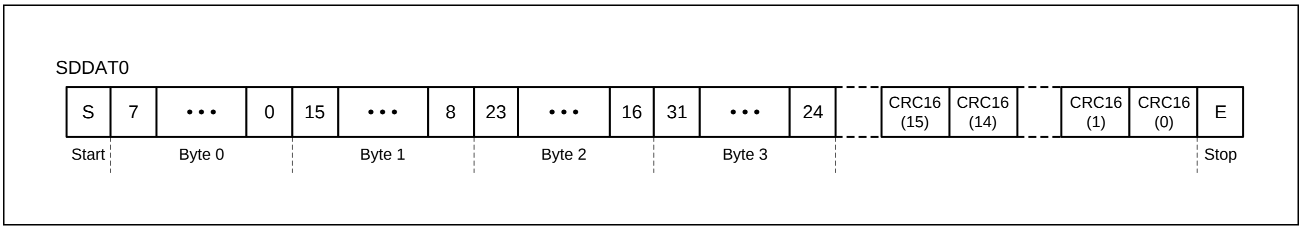


Figure 40.2 SDDAT in 1-Bit Width Mode

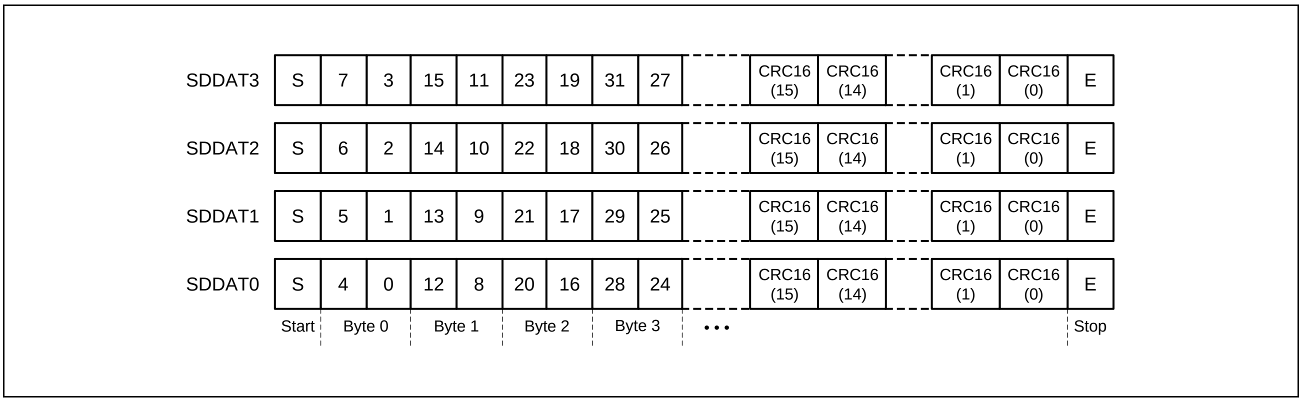


Figure 40.3 SDDAT in 4-Bit Width Mode

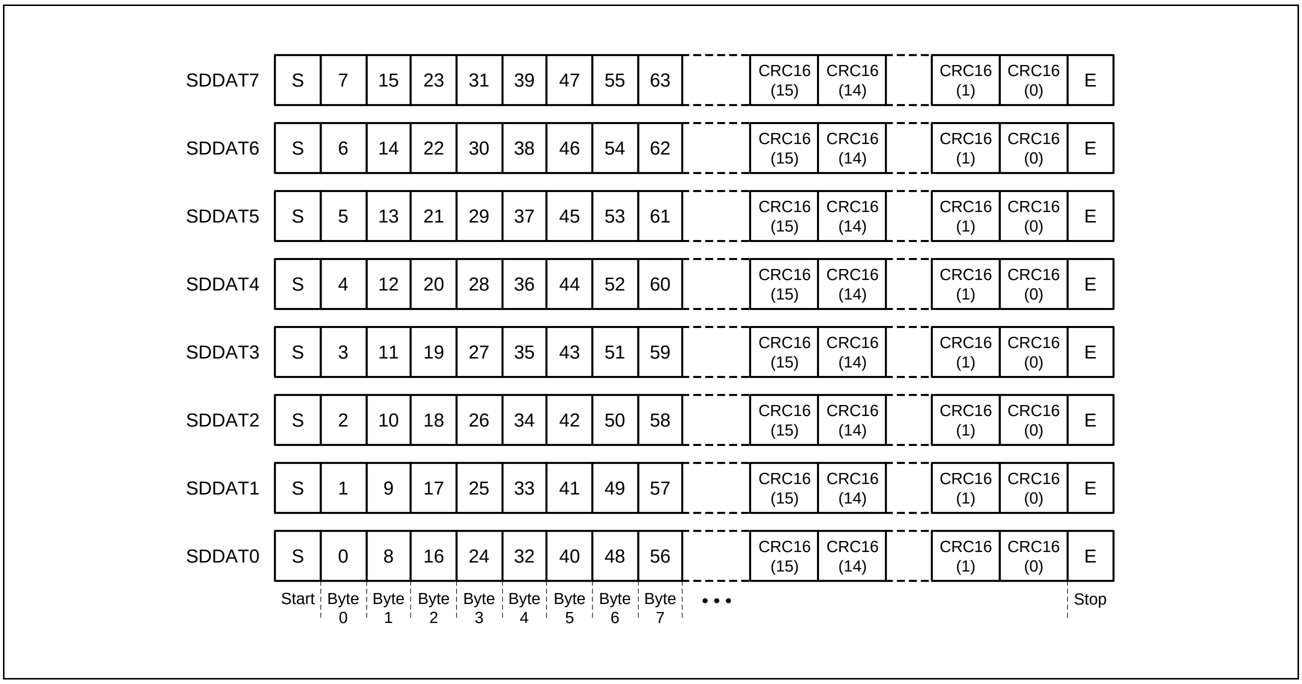


Figure 40.4 SDDAT in 8-Bit Width Mode

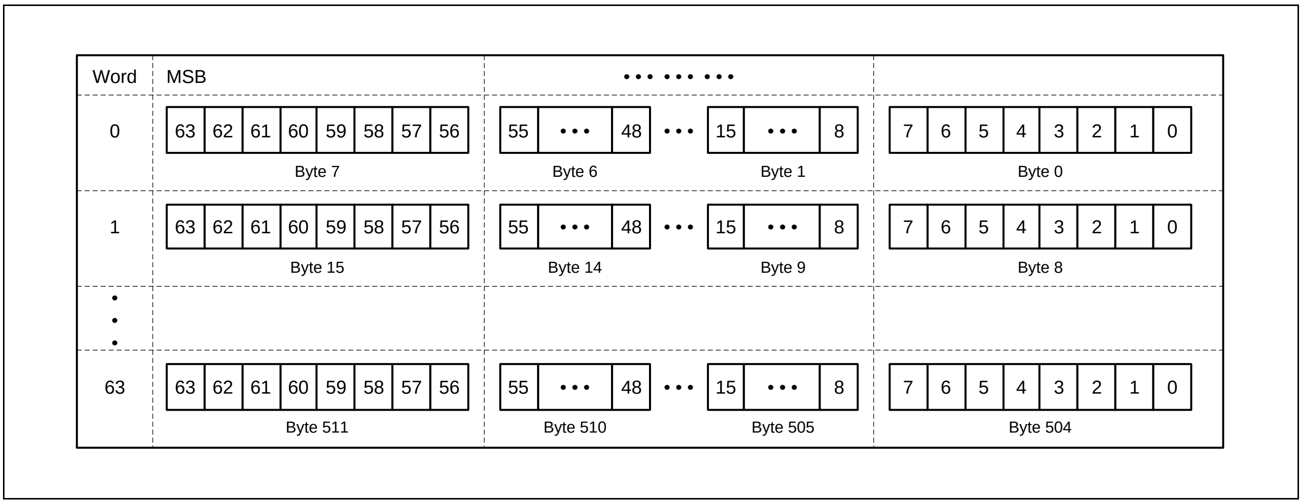


Figure 40.5 SD_BUF Store Data

Table 40.4 Reading from SD_BUF0

WMODE*[1]	BUSWIDTH*[1]	ENDIAN*[1]	Read Data*[2]
0	0	0	H’0123456789ABCDEF
0	0	1	H’EFCDAB8967452301
1	1	0	H’89ABCDEF (1st) H’01234567 (2nd)
1	1	1	H’EFCDAB89 (1st) H’67452301 (2nd)
1	0	0	H’CDEF (1st) H’89AB (2nd) H’4567 (3rd) H’0123 (4th)
1	0	1	H’EFCD (1st) H’AB89 (2nd) H’6745 (3rd) H’2301 (4th)

Note 1. The name of a bit in HOST_MODE.

Note 2. When the data stored in SD_BUF is H’0123456789ABCDEF

(2) Bus Signal Voltage Switch

Change the electric potential of the bus signal in the following procedure after checking that the SD card supports 1.8 V.

(1) Issuing CMD11
Perform command sequence processing of CMD11.

(2) Stopping the SD clock (a)
Set the SCLKEN bit in the SD_CLK_CTRL to 0 to stop*[1] the output of the SD clock. When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SDCLKOFFEN bit is also set to 0.

Note 1. When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SD clock has automatically been stopped.

(3) Checking the value of SDDAT
Check that the DAT0 bit in the SD_INFO2 register is 0.

(4) Changing the supply voltage of the host device
Change the voltage which is supplied through the power supply pin of the given channel (SD0_PVDD for channel 0 or SD1_PVDD for channel 1) from 3.3 V to 1.8 V.

For details, see the following sections:

Section 41.3.4, Port Function Control Register,
Section 41.3.7, Driving Ability Control Register,
Section 41.3.17, SD ch0 IO Voltage Mode Control Register,
Section 41.3.18, SD ch1 IO Voltage Mode Control Register.

(5) Starting supply of the SD clock (b)
After the SD clock has been stopped ((a) above) and 5 ms or more has elapsed, set the SCLKEN bit in the SD_CLK_CTRL register to 1 and allow the output of the SD clock. The SDCLKOFFEN bit must be 0.

(6) Checking the value of SDDAT
After supplying the SD clock has been started ((b) above) and 1 ms or more has elapsed, check that the DAT0 bit in the SD_INFO2 register is 1. It is possible to set the SDCLKOFFEN bit in the SD_CLK_CTRL register to 1 and allow SD Clock (SDCLK) Output Automatic Control Enable.

40.3.2 Card Detect/Write Protect

(1) Card Detect

The SD/MMC host interface has two types of card detect functions as described in the following.

- Card detect with ISDCD

  Figure 40.6 shows the timing chart of card detect using ISDCD. ISDCD is connected to the card socket and pulled up on the host device. The resistance of the pull-up resistor is decided by the specification of the SD host device.

  [Card insertion]

  ISDCD is pulled down when a card is inserted. At this time, if ISDCD has been pulled down for the Mcycle period (set in SD_OPTION), INFO4 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

  [Card removal]

  ISDCD is pulled up when a card is removed. At this time, if ISDCD has been pulled up for the Mcycle period (set in SD_OPTION), INFO3 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

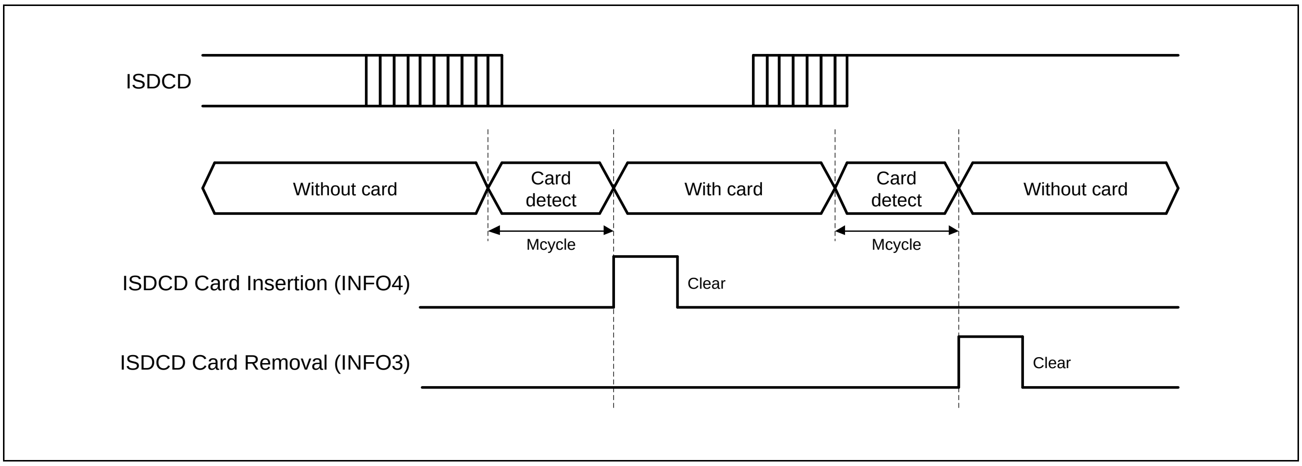


Figure 40.6 Example of Card Detect with ISDCD

- SD card detect with SDDAT3

  Figure 40.7 shows the timing chart when the SD card is detected using SDDAT3. In addition, SDDAT3 is pulled down on the host device. The resistance of the pull-down resistor is decided by the specification of the SD host device.

  [Card insertion]

  When an SD card is inserted, SDDAT3 is pulled up. Accordingly, INFO9 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

  [Card removal]

  When an SD card is removed, SDDAT3 is pulled down. Accordingly, INFO8 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

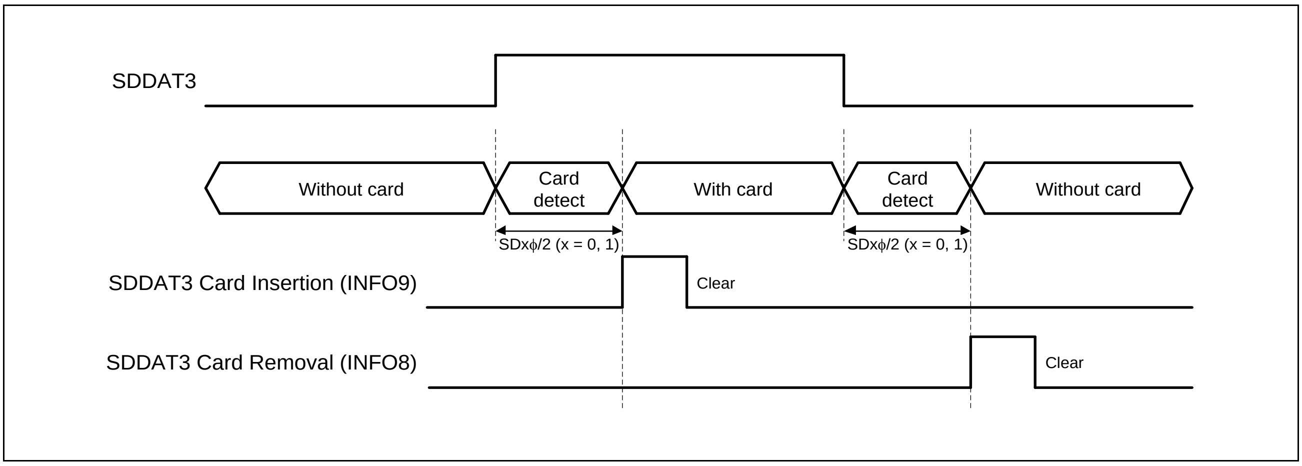


Figure 40.7 SD Card Detect with SDDAT3

(2) Write Protect

The SD/MMC host interface has two types of write protect functions.

- Write protect with ISDWP

  ISDWP is connected to the card socket, and pulled up or pulled down by the card insertion. The selection of pulling up or pulling down and the resistance value is decided by the specifications of the SD host device. As the ISDWP state is reflected to INFO7 in SD_INFO1, the write protect is decided after the SD card is inserted.

- Write protect with command

  The card’s internal write protection and the card lock/unlock operation are realized by the command.

40.3.3 Interrupt Request

(1) Interrupt Request

The SD/MMC host interface has the interrupt requests shown in **Table 40.5** shows the relationship between the interrupt flag registers and the interrupt mask registers. When a bit in an interrupt mask register is set to 0, an interrupt occurs by setting the corresponding bit in the interrupt flag register to 1.

To clear a flag, write 0 to the bit to be cleared and 1 to the other bits.

Table 40.5 Interrupt Request

	Interrupt Flag Register		Interrupt Mask Register	
Interrupt Request	Register Name	Bit Name	Register Name	Bit Name
Card access interrupt*[1]	SD_INFO1	INFO2	SD_INFO1_MASK	IMASK2
		INFO0		IMASK0
	SD_INFO2	ILA	SD_INFO2_MASK	IMASK
		BWE		BMASK1
		BRE		BMASK0
		ERR6		EMASK6
		ERR5		EMASK5
		ERR4		EMASK4
		ERR3		EMASK3
		ERR2		EMASK2
		ERR1		EMASK1
		ERR0		EMASK0
SDIO access interrupt*[2]	SDIO_INFO1	EXWT	SDIO_INFO1_MASK	MEXWT
		EXPUB52		MEXPUB52
		IOIRQ		IOMSK
Card detect interrupt*[1]	SD_INFO1	INFO9	SD_INFO1_MASK	IMASK9
		INFO8		IMASK8
		INFO4		IMASK4
		INFO3		IMASK3
DMAC interrupt*[1]	DM_CM_INFO1	DTRANEND1	DM_CM_INFO1_MASK	DTRANEND1_MASK
		DTRANEND0		DTRANEND0_MASK
		SEQEND		SEQEND_MASK
	DM_CM_INFO2	DTRANERR1	DM_CM_INFO2_MASK	DTARERR1_MASK
		DTRANERR0		DTARERR0_MASK
		SEQERR		SEQERR_MASK

Note 1. Interrupt signal “OXMNIRQ” is asserted.

Note 2. Interrupt signal both “OXMNIRQ” and “OXASIOIRQ” are asserted.

40.3.4 Communications Errors and Timeouts

- Communications Errors and Timeouts

  Table 40.6 and **Table 40.7** show the relationships between the SD card interrupt flag register and SD error status register for communications errors and timeouts, respectively. When a bit in the SD card interrupt flag register is set to 1, the corresponding bit in the SD error status register is set to 1. The values of the SD error status register are cleared by writing to SD_CMD or writing 0 to the SDRST bit in SOFT_RST.

Table 40.6 Communications Errors

Communication Error	SD Interrupt Flag Register		SD Error Status Register		Description
	Register Name	Bit Name	Register Name	Bit Name	
END error	SD_INFO2	ERR2	SD_ERR_STS1	E5	When an error occurs in the CRC status length
				E4	When an error occurs in read data length
				E3	When an error occurs in the response length to a command issued within a command sequence
				E2	When an error occurs in the response length (other than a response to a command issued within a command sequence)
CRC error		ERR1		E11	When an error occurs in the CRC status
				E10	When a CRC error occurs in the read data
				E9	When a CRC error occurs in the response to a command issued within a command sequence
				E8	When a CRC error occurs in the response (other than a response to a command issued within a command sequence)
CMD error		ERR0		E1	The command index of the transmitted command differed from the command index of the received response (for a command issued within a command sequence)
				E0	The command index of the transmitted command differed from the command index of the received response (for a command issued other than within a command sequence)

Table 40.7 Timeouts

Timeout	SD Interrupt Flag Register		SD Error Status Register		Description
	Register Name	Bit Name	Register Name	Bit Name	
Response timeout	SD_INFO2	ERR6	SD_ERR_STS2	E1	When the response to a command issued within a command sequence is not received even after 640 cycles of SDCLK have elapsed
				E0	When the response (other than a response to a command issued within a command sequence) is not received even after 640 cycles of SDCLK have elapsed
Data timeout (other than response timeout)		ERR3		E6	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*[1] after the CRC status
				E5	When the CRC status is not received t even after Ncycle*[1] has elapsed after data writing
				E4	When read data is not received even after Ncycle*[1] has elapsed after read command
					When read data for the next block are not received even after Ncycle*[1] has elapsed after the reception of read data
					When read data for the next block are not received even after Ncycle*[1] has elapsed after release of the read wait state
				E3	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*[1] after CMD12 has been issued within a command sequence
				E2	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*[1] after R1b response

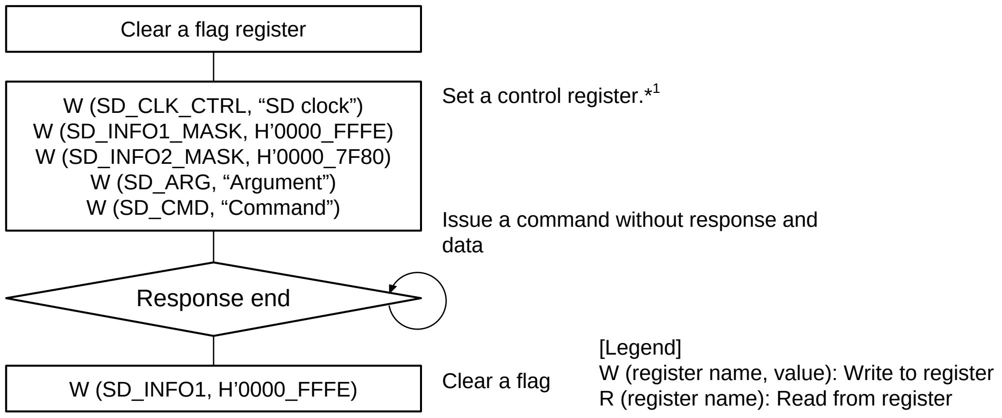
Note 1. Ncycle is set by bit 7 to bit 4 in SD_OPTION.

40.4 Usage Example

40.4.1 Command without Data Transfer

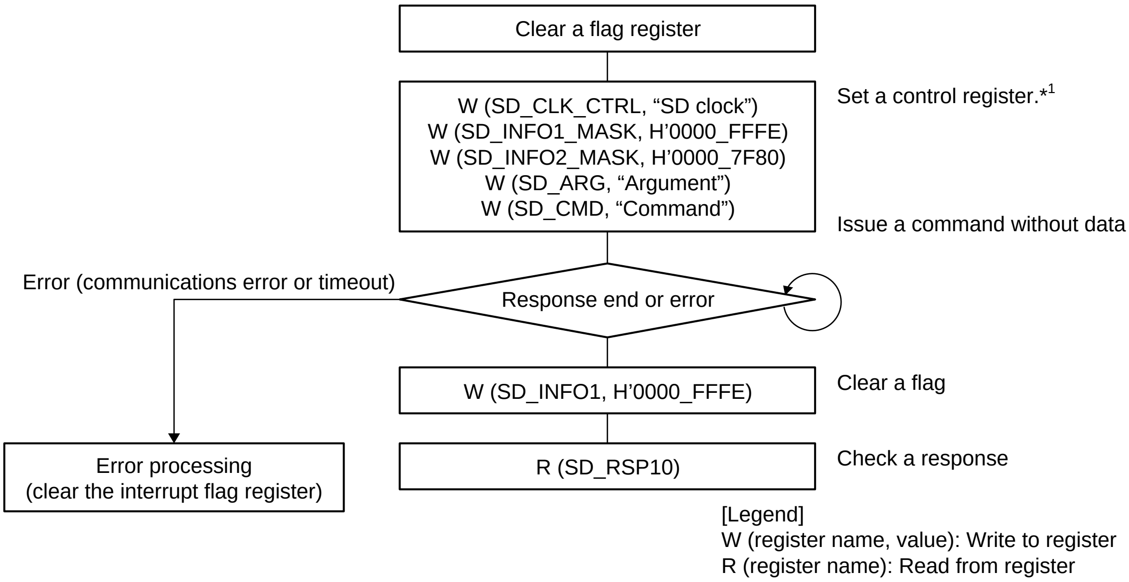
(1) Flowchart

Figure 40.8 and **Figure 40.9** show flowchart examples.



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.8 Flow Example of Command without Response and Data



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.9 Flow Example of Command without Data

(2) Operation for Command without Data Transfer

The legend below is used for description of register read/write.

W (register name, value):Write to register

R (register name):Read from register

The operation is described as below.

(a) Command without response and data

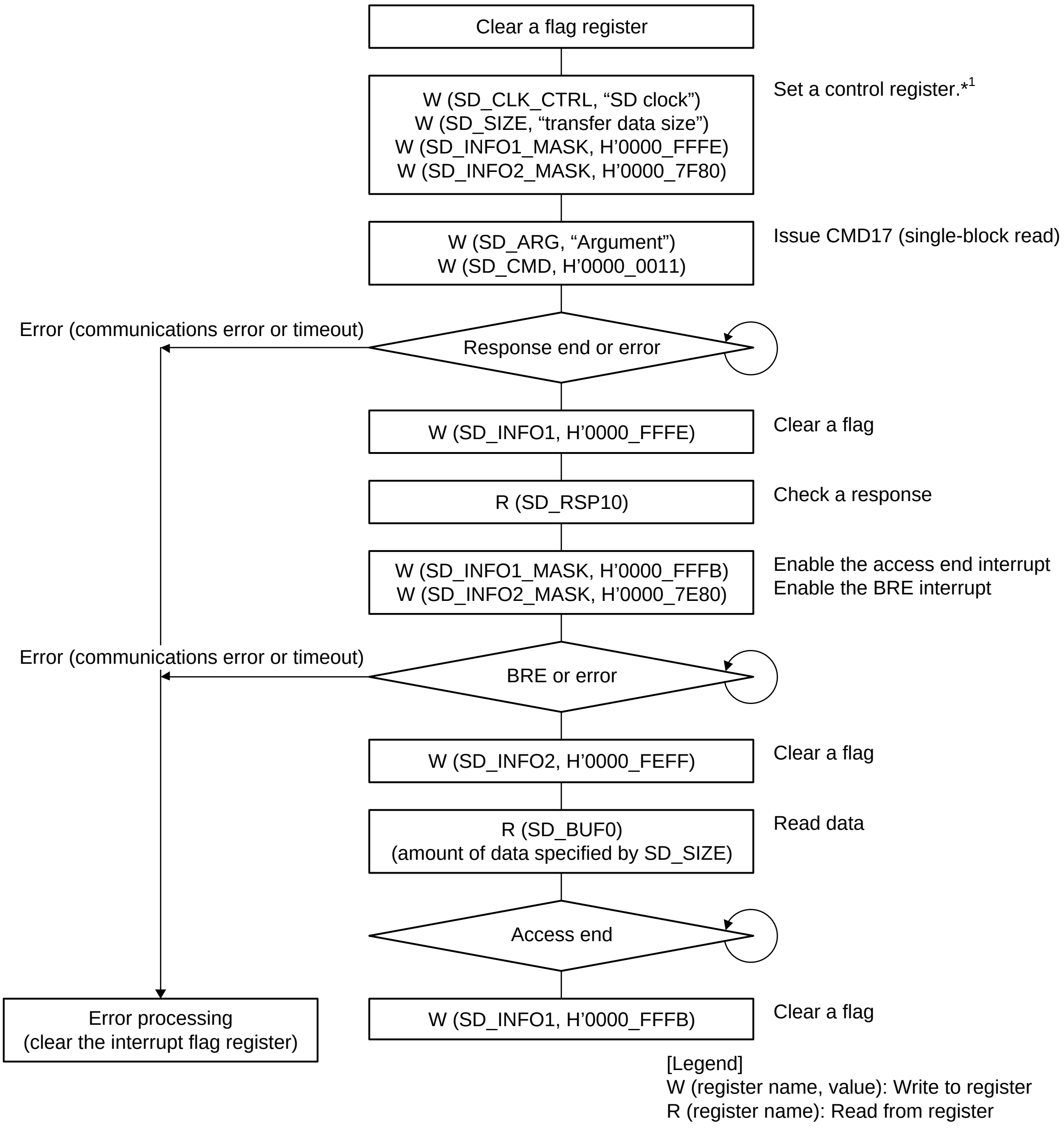
1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   When transmission of a command is completed, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.

(b) Command without data

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   When a response is received, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.
5. Read a response from SD_RSP10.
   Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

40.4.2 Single Block Read

(1) Flowchart



Note: The error processing of SDR104/HS200 requires a re-tuning.

Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.10 Single Block Read Flowchart Example

(2) Operation for Single Block Read

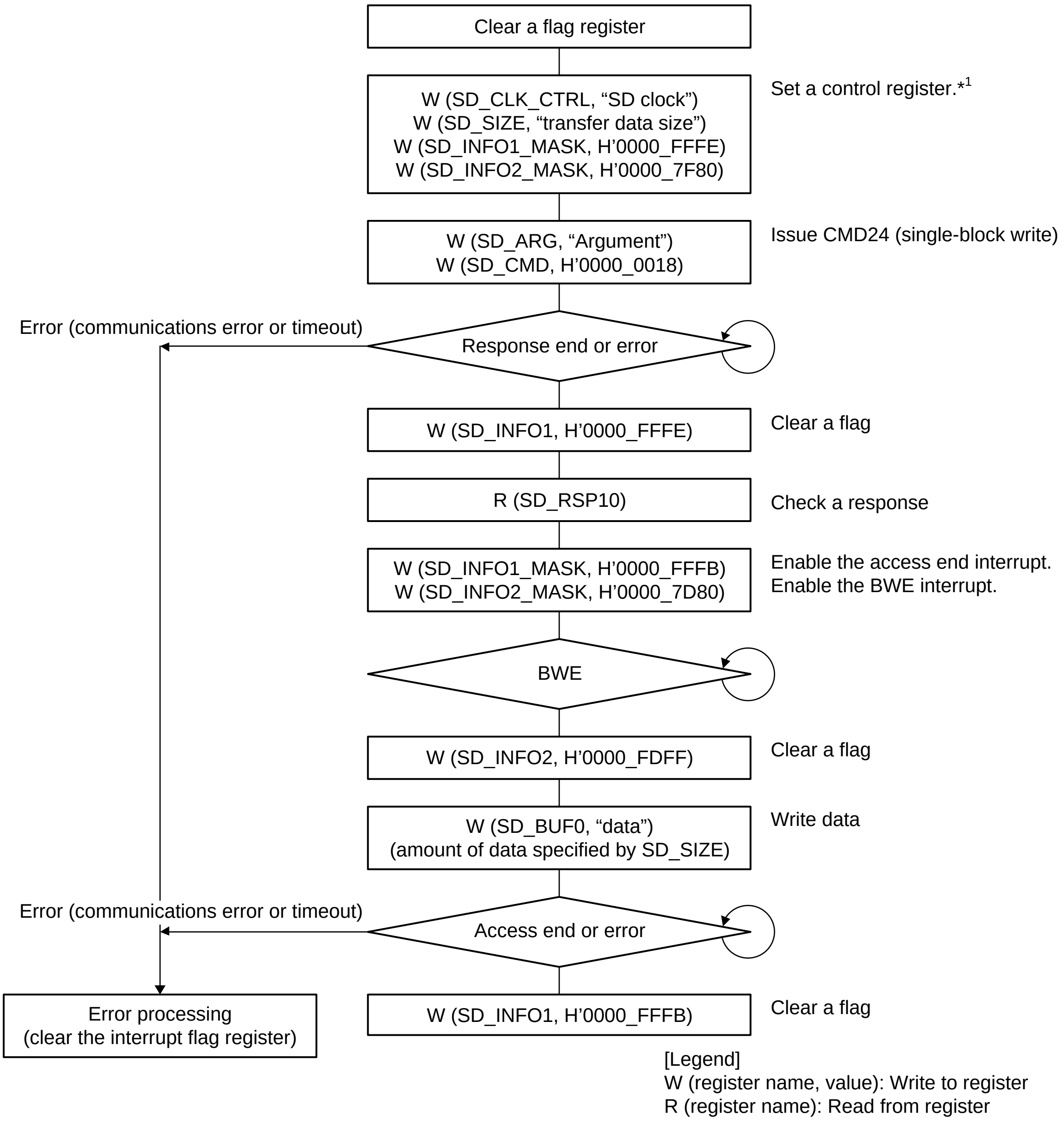
The operation of the single block read is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD17)
   Set CMD17 Argument in SD_ARG and write H’0000 0011 to SD_CMD.
   Accordingly, CMD17 is issued, and the single block read operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data receive from SD card and data read
   Write H’0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H’0000 7E80 to SD_INFO2_MASK to enable the BRE interrupt. When the data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0.
   However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress.
6. Operation complete
   When the data read from SD_BUF0 is completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to end the single block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

40.4.3 Single Block Write

(1) Flowchart



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.11 Single Block Write Flowchart Example

(2) Operation for Single Block Write

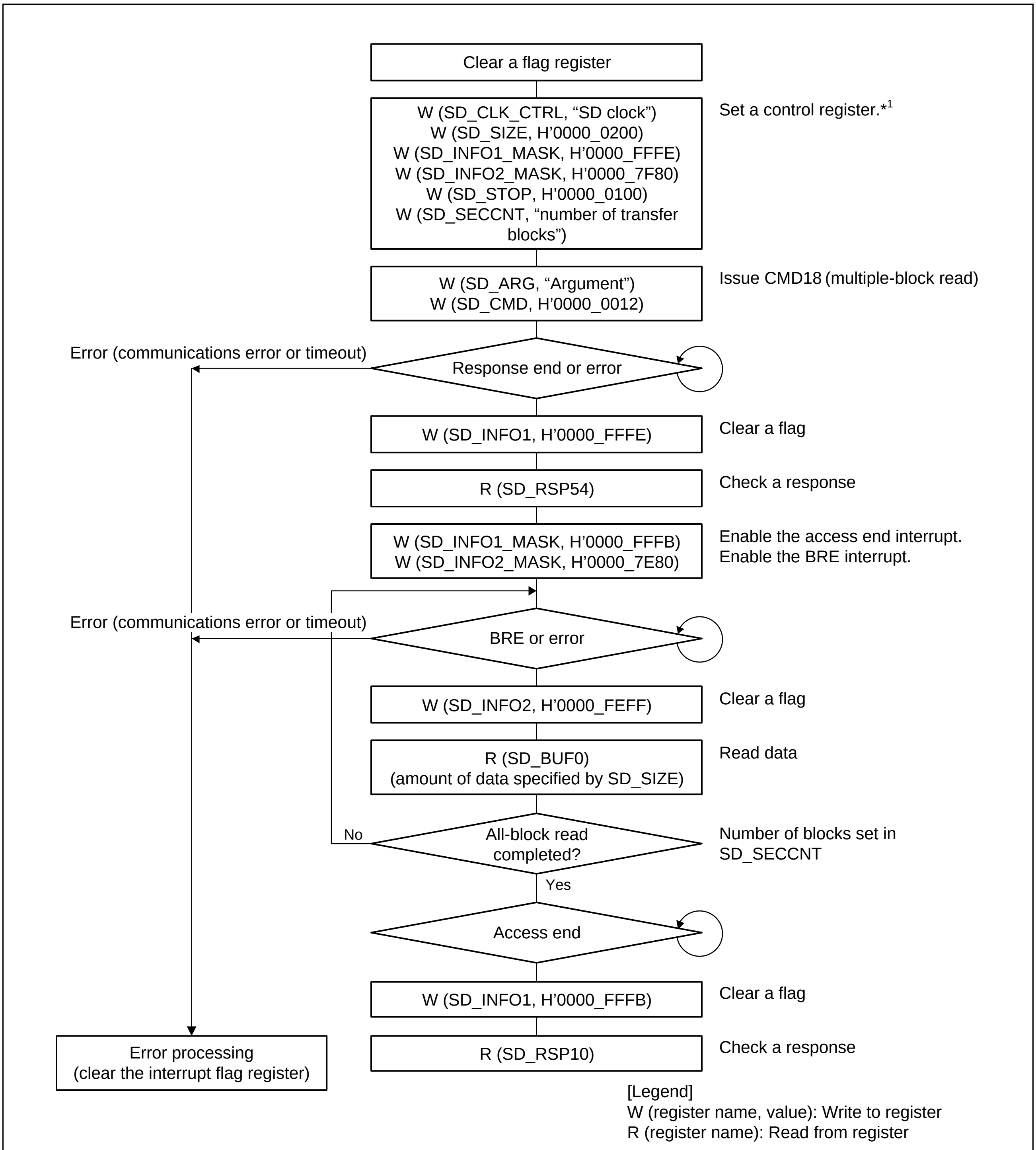
The operation of the single block write is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24)
   Set CMD24 Argument in SD_ARG and write H’0000 0018 to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card
   Write H’0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H’0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted after writing to SD_BUF0.
6. Operation complete
   When the CRC status and busy state are received from the SD card, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to end the single block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

40.4.4 Multiple Block Read

(1) Flowchart



Note: The error processing of SDR104/HS200 requires a re-tuning.

Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.12 Multiple Block Read Flowchart Example

(2) Operation for Multiple Block Read

The operation of the multiple block read is described below.

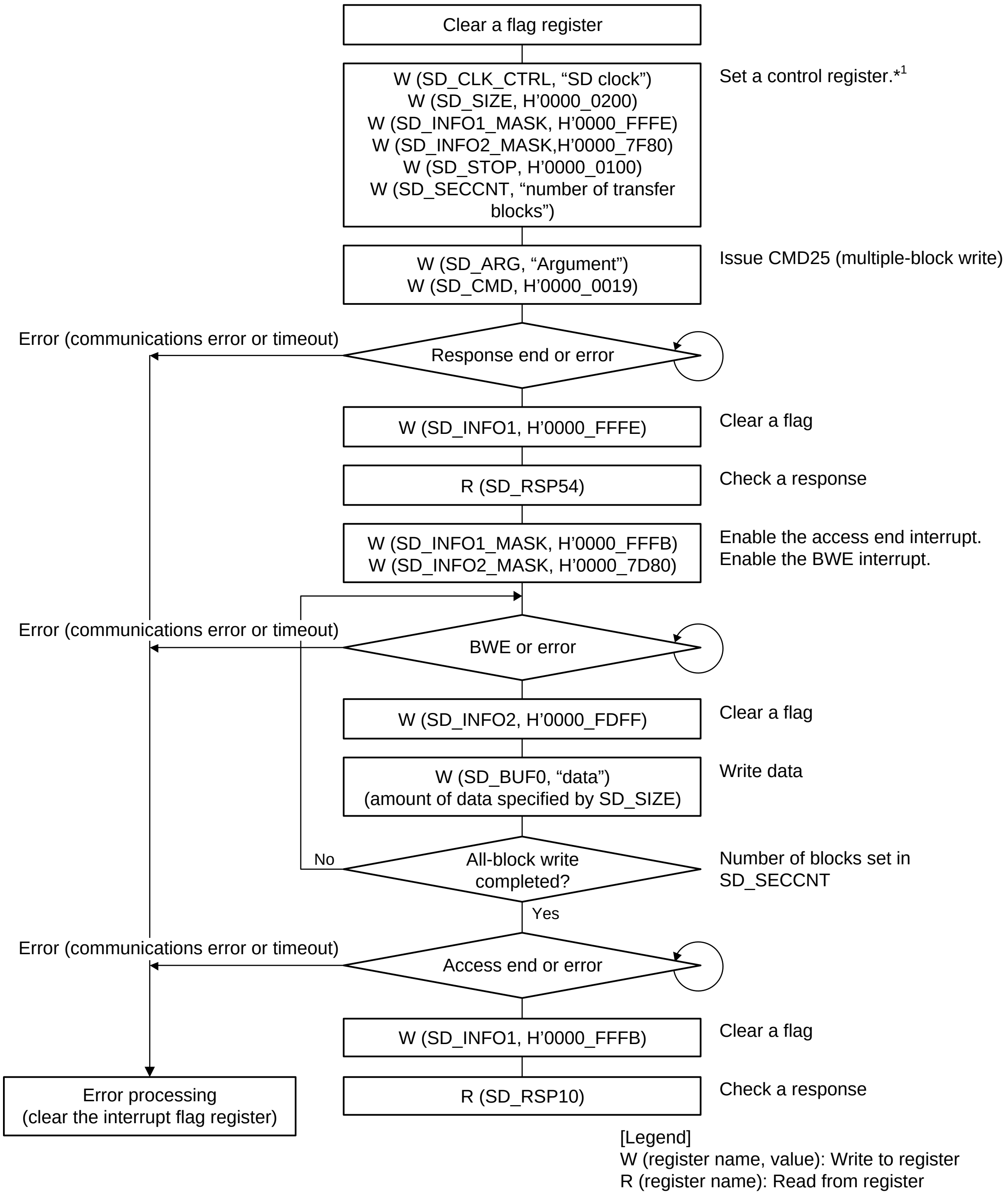
1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD18)
   Set CMD18 Argument in SD_ARG and write H'0000 0012 to SD_CMD.
   Accordingly, CMD18 is issued, and the multiple block read operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data receive from SD card and data read
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7E80 to SD_INFO2_MASK to enable the BRE interrupt. When one-block data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to H'0000 000.
6. Operation complete
   When all-block data read and the CMD12 response receive are completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to read the response. This is the end of multiple block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

40.4.5 Multiple Block Write (when Using Internal Timer)

(1) Flowchart

Figure 40.13 shows the flowchart when using an internal timer.



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.13 Multiple Block Write Flowchart Example (when Using Internal Timer)

(2) Operation for Multiple Block Write

The operation of the multiple block write is described below.

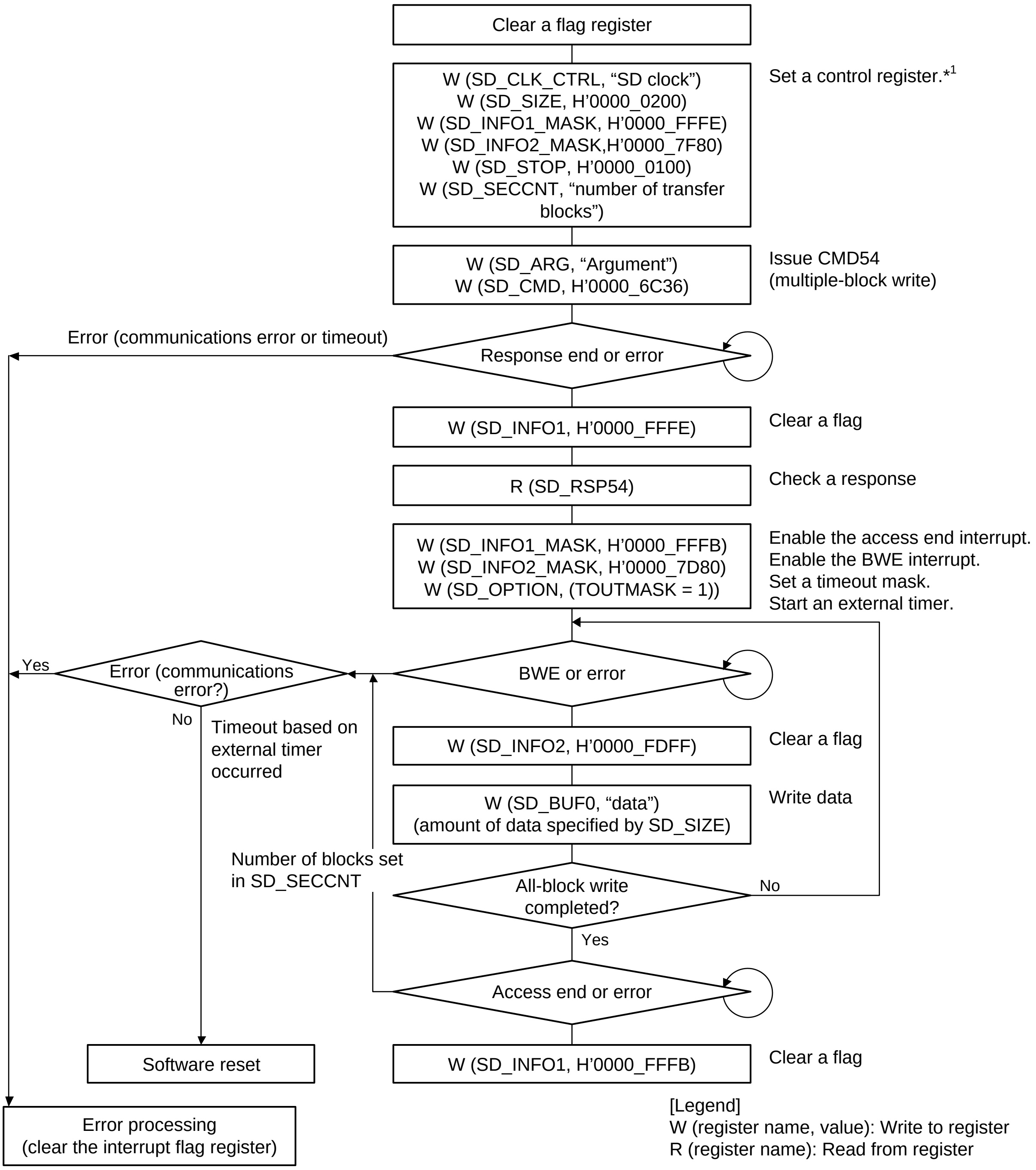
1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write H’0000 0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H’0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H’0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to H’0000 0000.
6. Operation complete
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

40.4.6 Multiple Block Write (when Using External Timer)

(1) Flowchart

The flowchart when using an external timer instead of an internal timer of this module is shown below.



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.14 Multiple Block Write Flowchart Example (when Using External Timer)

(2) Operation for Multiple Block Write

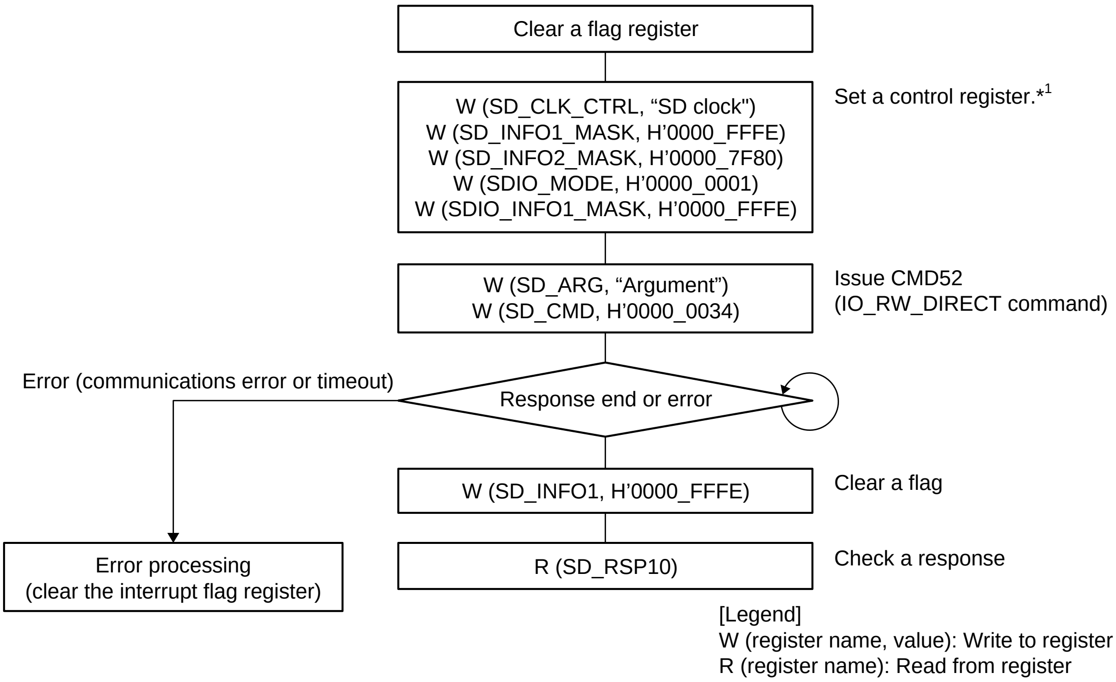
The operation of the multiple block write is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD54)
   Set CMD54 Argument in SD_ARG and write H'0000 6C36 to SD_CMD.
   Accordingly, CMD54 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H'0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H'0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. Set the TOUTMASK bit in SD_OPTION to disable timeout and start an external timer.
   When SD_BUF0 is ready for the data to be written, the BWE bit in the SD_INFO2 resister is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Operation complete
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs at response reception (a communications error or timeout) or at data transmission. Perform a software reset if a timeout occurs at data transmission based on an external timer.

40.4.7 IO_RW_DIRECT Commend (CMD52)

(1) Flowchart



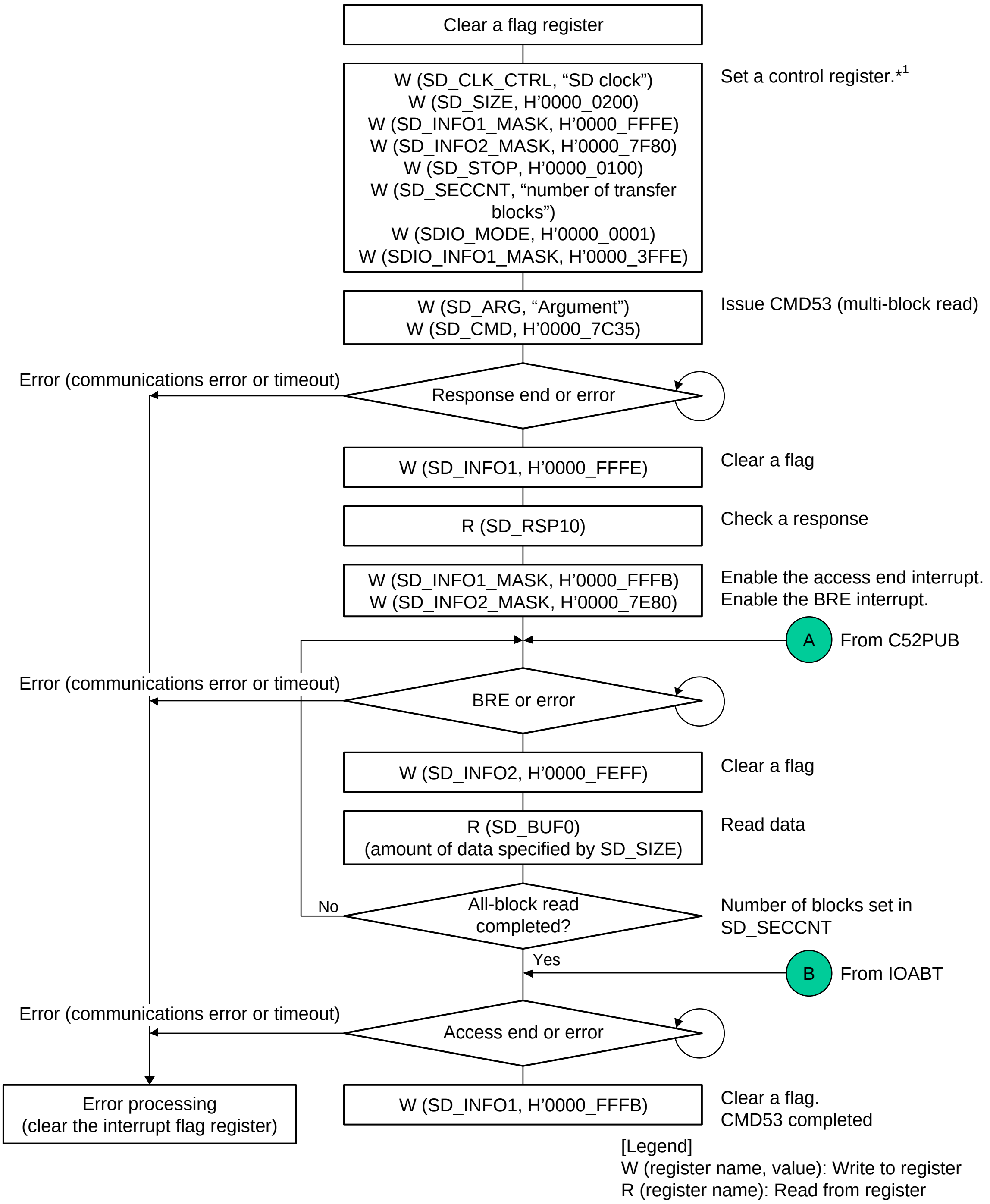
Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.15 IO_RW_DIRECT Command (CMD52) Flowchart Example

40.4.8 IO_RW_EXTENDED (CMD53/Multiple Block Read)

(1) Flowchart

Figure 40.16 shows a flowchart example for CMD53 (multiple block read).



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.16 CMD53 (Multiple Block Read) Flowchart Example

Figure 40.17 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block read).

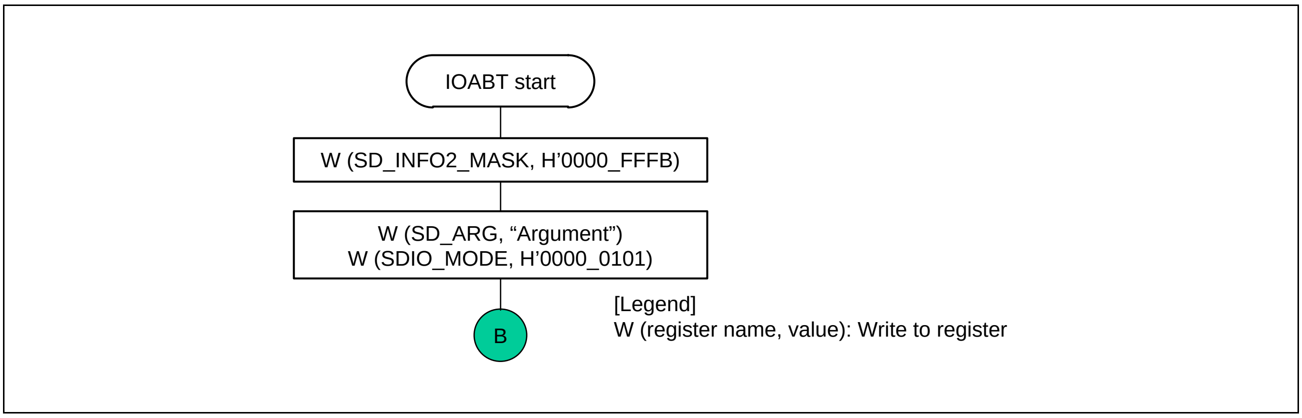


Figure 40.17 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Read)

Figure 40.18 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block read) while the SD/MMC host interface is in the read wait state.

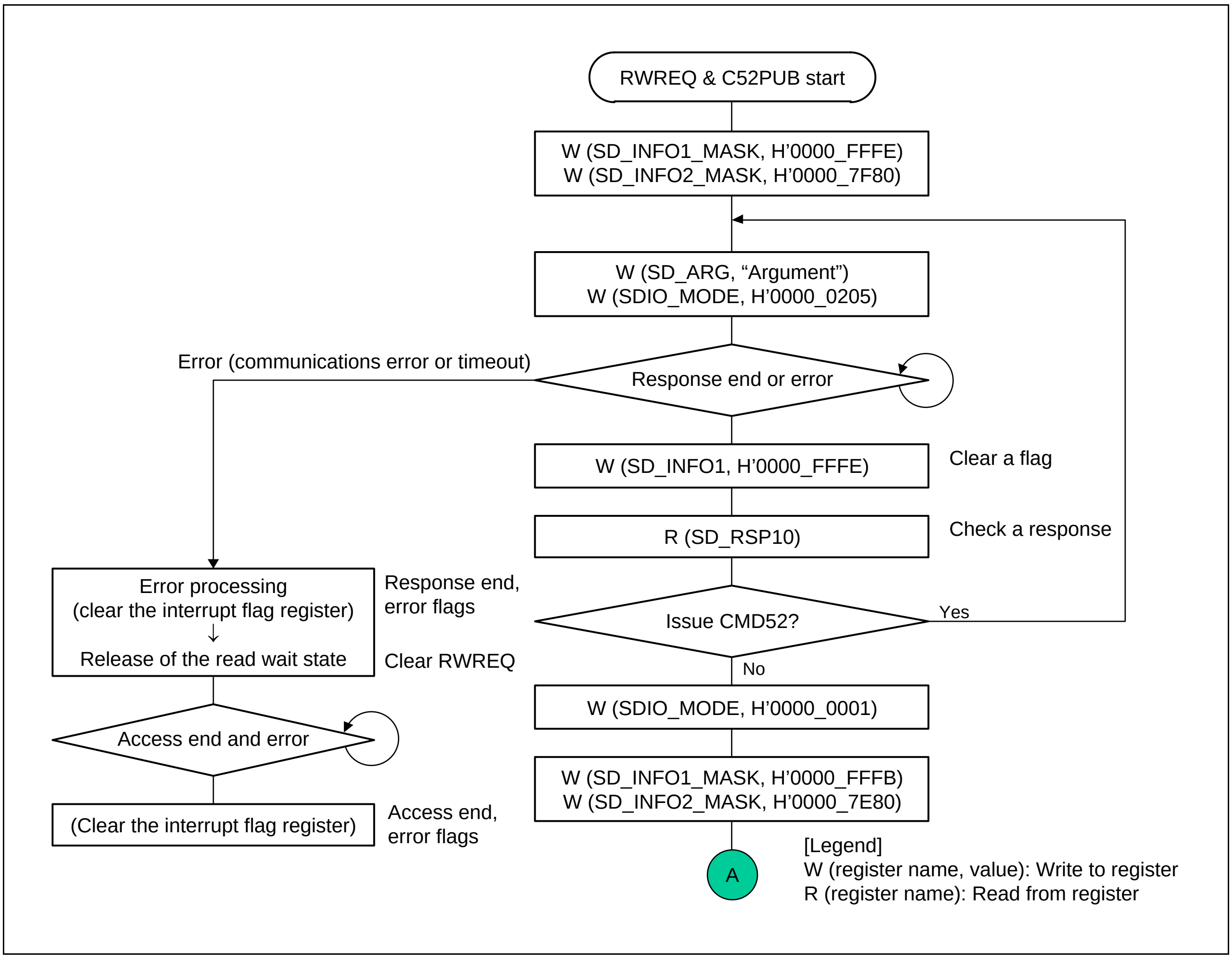
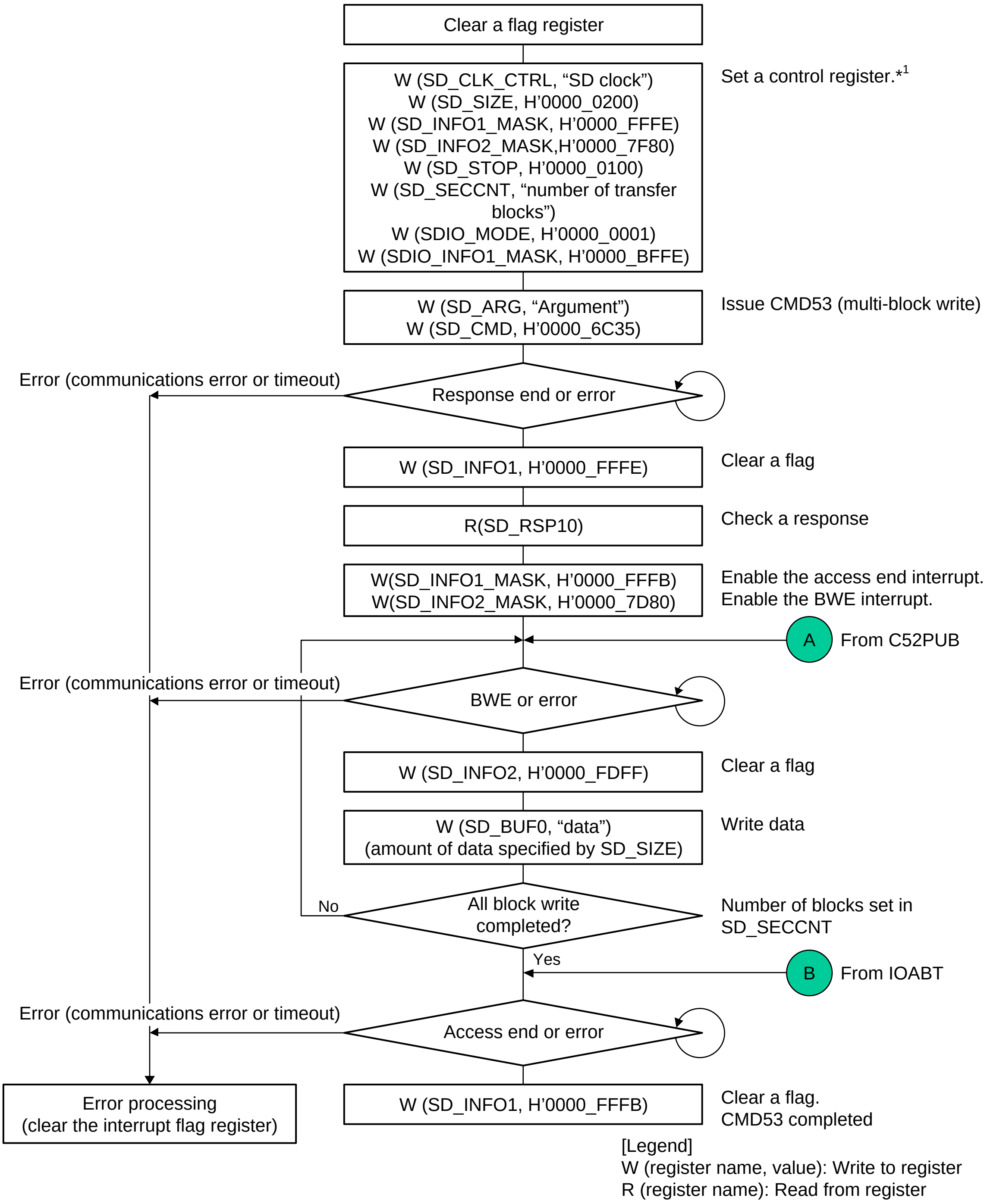


Figure 40.18 Flowchart Example when CMD52 (SDIO None Abort) is Issued after Read Wait State is Entered at CMD53 (Multi Block Read)

40.4.9 IO_RW_EXTENDED (CMD53/Multiple Block Write)

(1) Flowchart

Figure 40.19 shows a flowchart example for CMD53 (multiple block write).



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.19 CMD53 (Multiple Block Write) Flowchart Example

Figure 40.20 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block write).

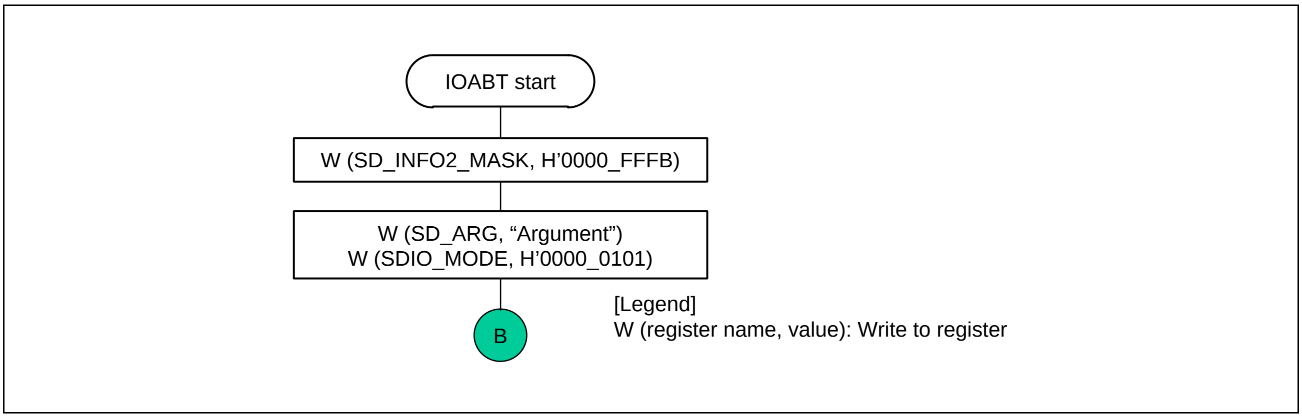


Figure 40.20 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Write)

Figure 40.21 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block write).

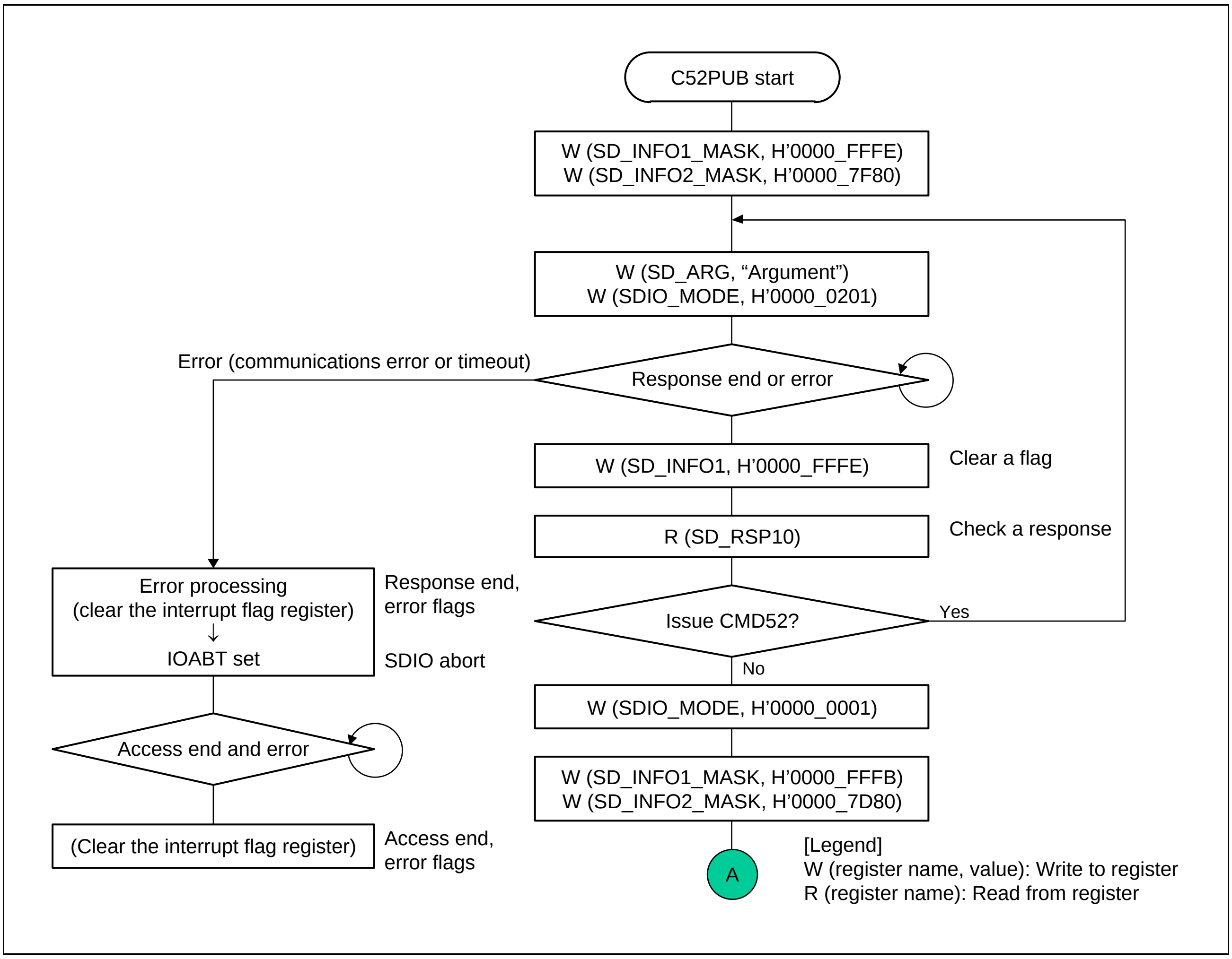
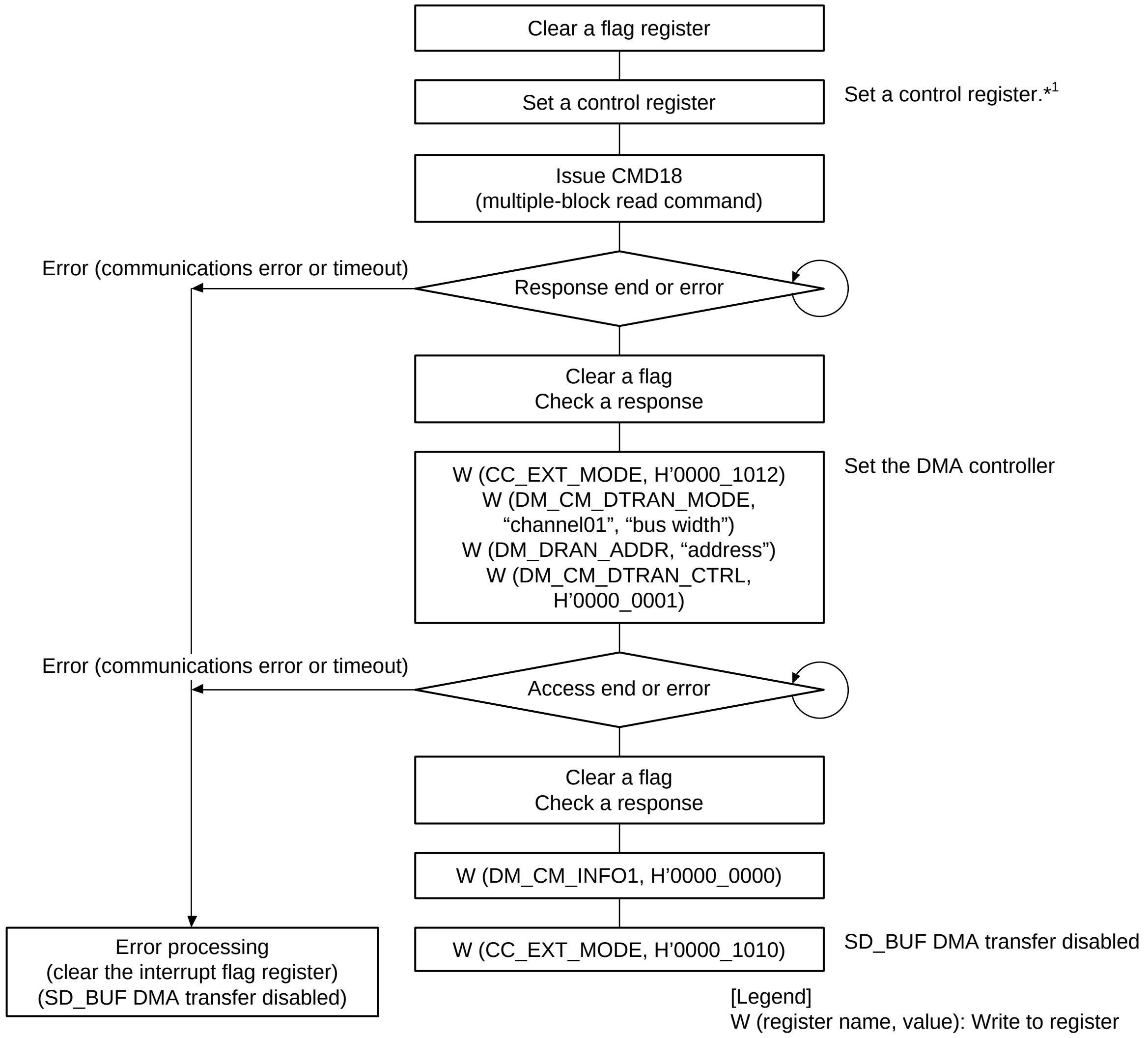


Figure 40.21 Flowchart Example when CMD52 (SDIO None Abort) is Issued at CMD53 (Multiple Block Write)

40.4.10 DMA Transfer

(1) SD_BUF DMA Transfer

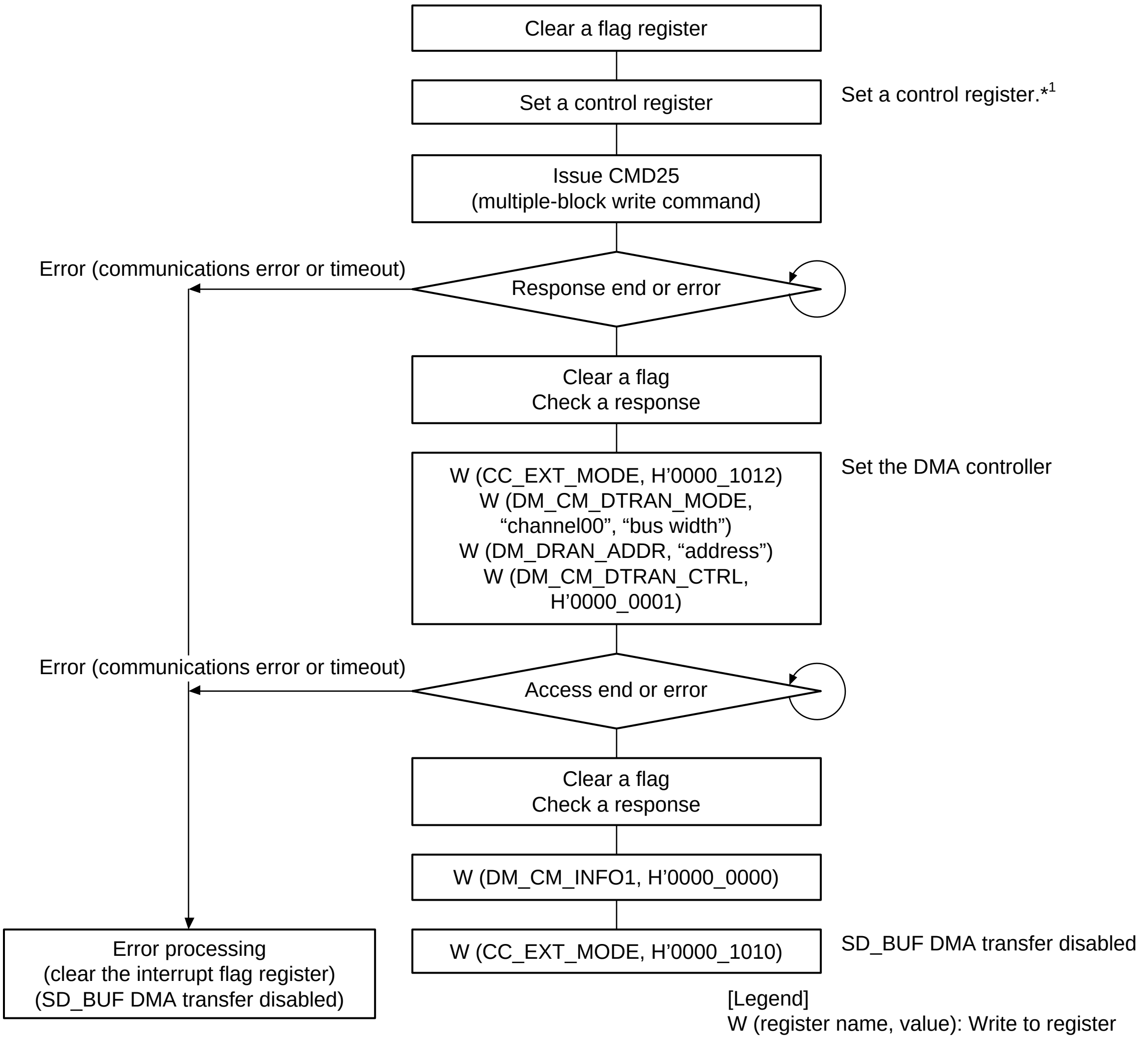
Figure 40.22 shows a flowchart example for SD_BUF DMA read when CMD18 (multiple block read) is issued.



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.22 SD_BUF DMA Read Flowchart Example

Figure 40.23 shows a flowchart example for SD_BUF DMA write when CMD25 (multiple block write) is issued.

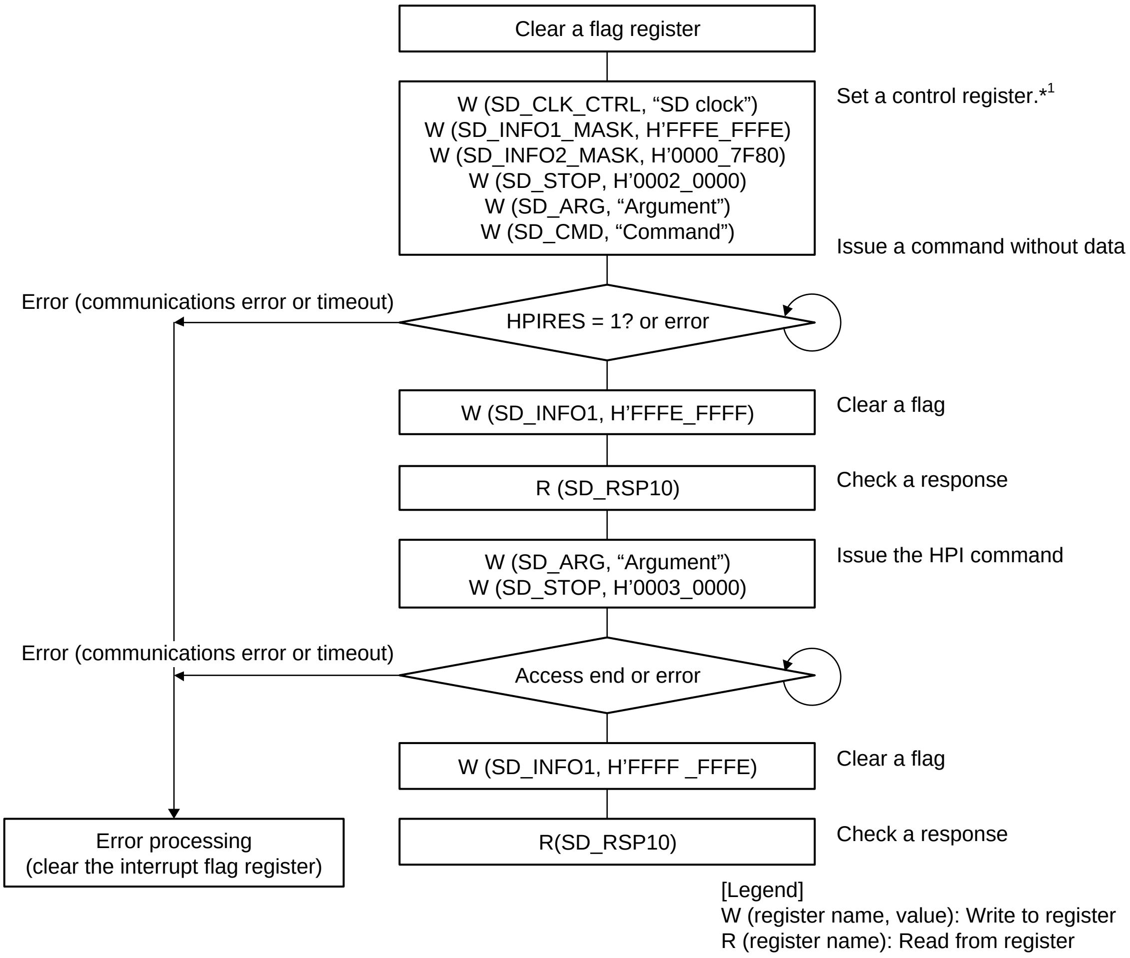


Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.23 SD_BUF DMA Write Flowchart Example

40.4.11 High-Priority Interrupt (without Data Transfer)

(1) Flowchart



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.24 Example of the High-Priority Interrupt (without Data Transfer) Flowchart

(2) Operation for High-Priority Interrupt without Data Transfer

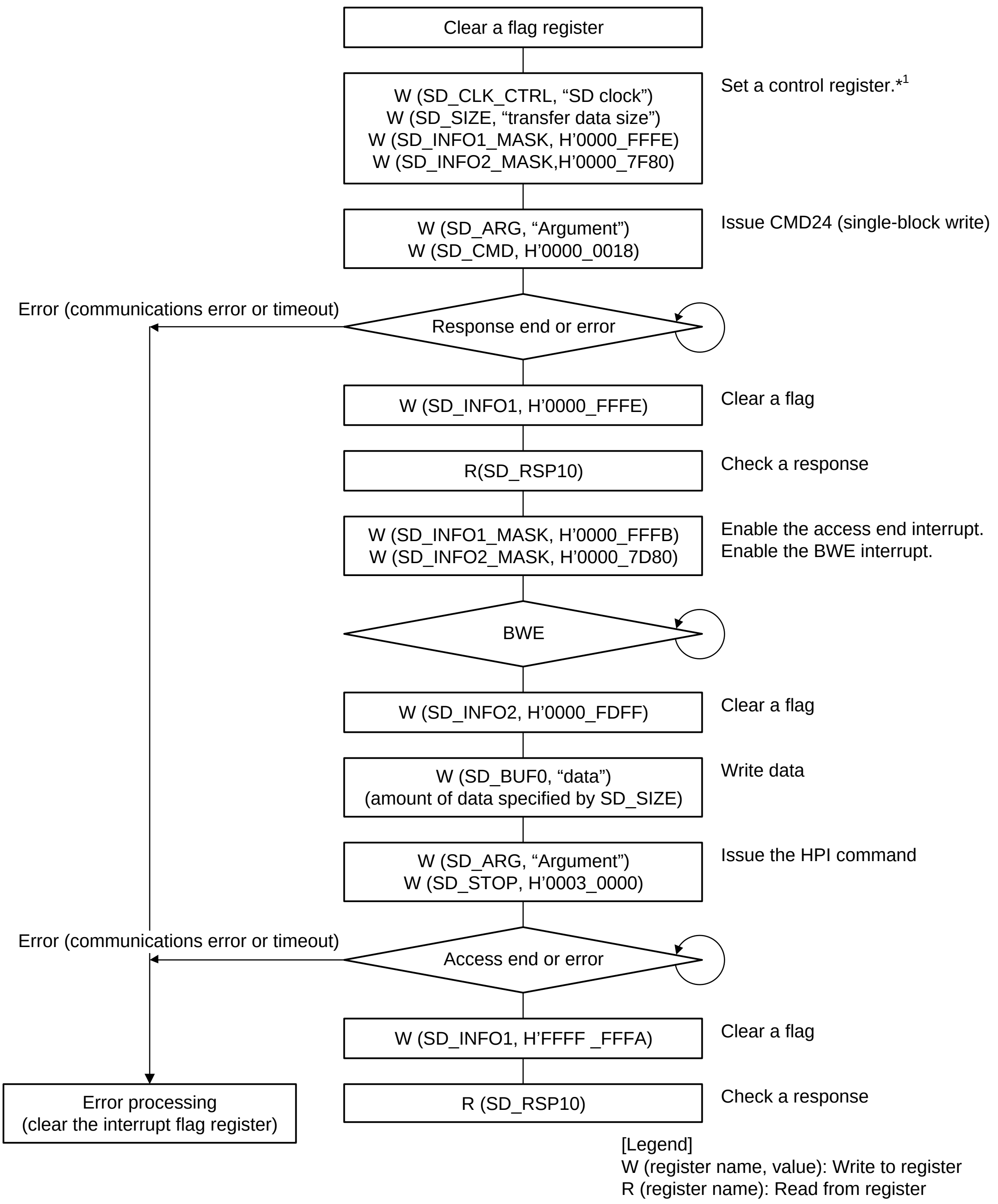
The operation of the high-priority interrupt (HPI) without data transfer is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), HPI enable, interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set the CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   On receiving the response, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0.
5. Read the response from SD_RSP10.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO0 bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO0 bit to 0 to read the response from SD_RSP10.
   This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

40.4.12 High-Priority Interrupt (at Single Block Write)

(1) Flowchart



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.25 Example of the High-Priority Interrupt (at Single Block Write) Flowchart

(2) Operation for HPI at Single Block Write

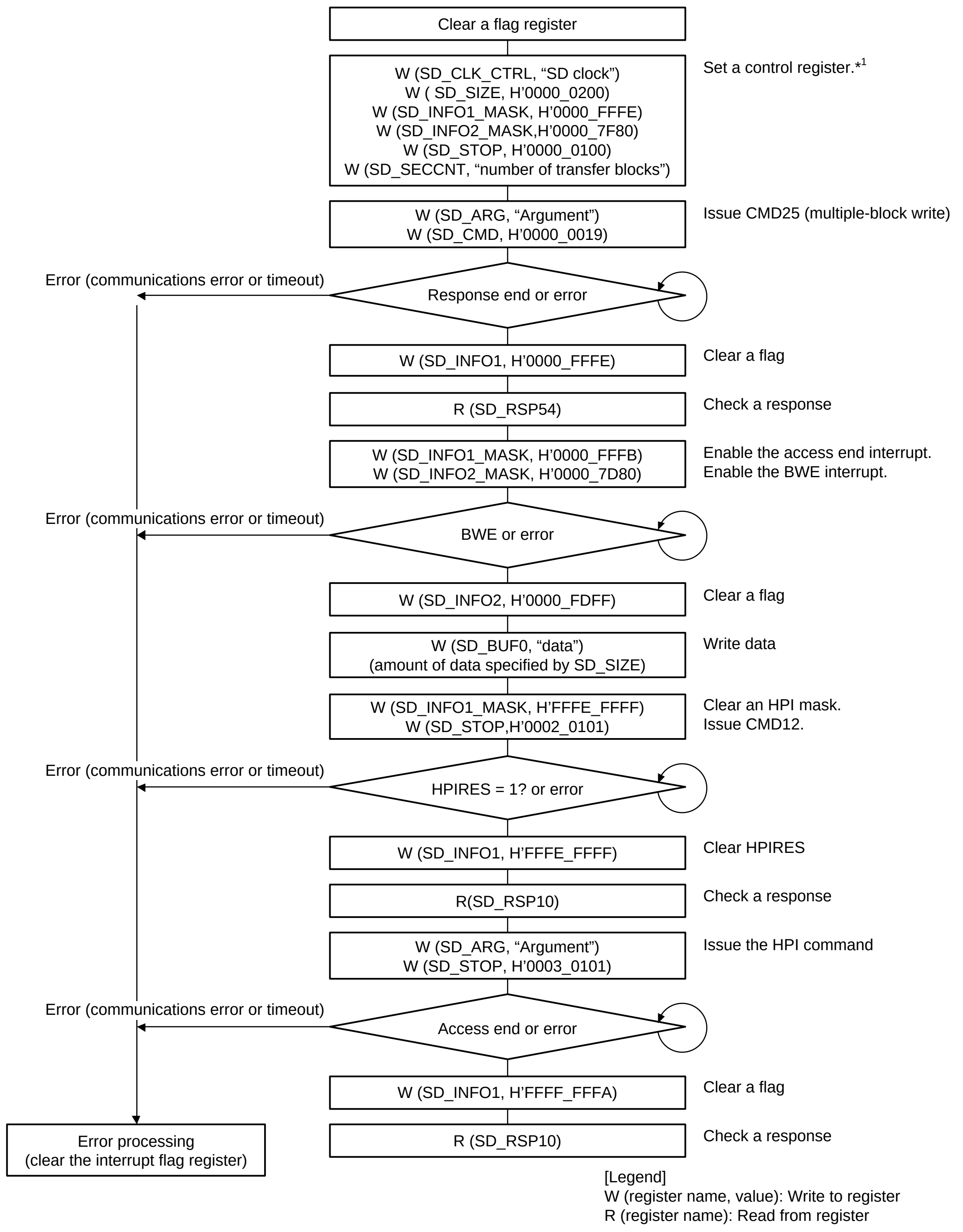
The operation of HPI at single block write is described below.

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24)
   Set CMD24 Argument in SD_ARG and write H’0000 0018 to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card
   Write H’0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H’0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

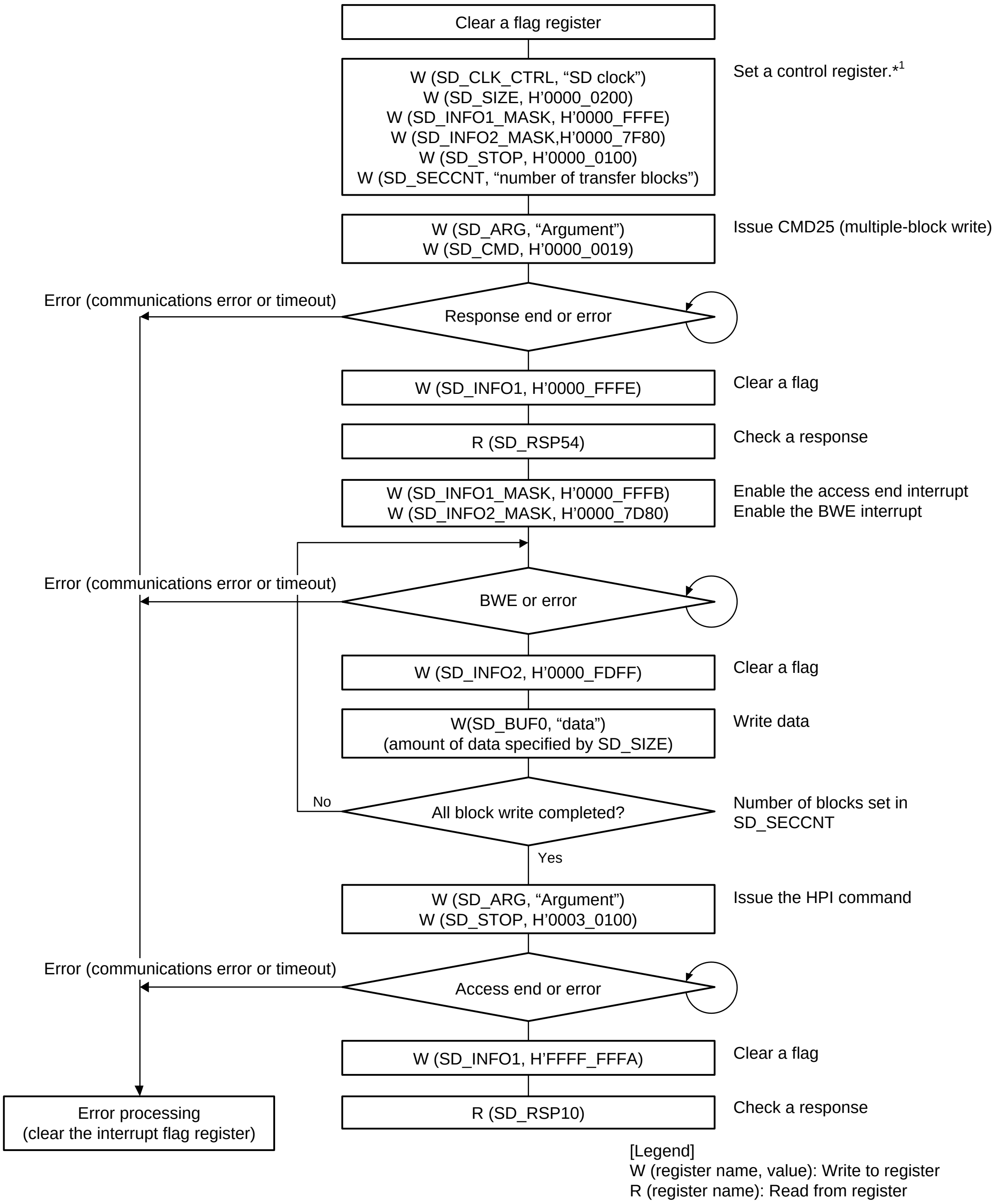
40.4.13 High-Priority Interrupt (at Multiple Block Write)

(1) Flowchart



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.26 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (a)



Note 1. Refer to **(5)** in **Section 40.5, Usage Notes**.

Figure 40.27 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (b)

(2) Operation for HPI at Multiple Block Transfer

The operation of HPI at the multiple block write is described below.

(a) When not all the data has been written to SD_BUF

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write H’0000 0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H’0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H’0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
   Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Command issue (CMD12)
   Write H’FFFE FFFF to SD_INFO1_MASK to enable the HPIRES interrupt. Write H’0002 0101 to SD_STOP, which causes CMD12 to be issued.
7. Response check
   When the response is received, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0 to read the response from SD_RSP10.
8. HPI command issue (CMD25)
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
9. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

(b) When all the data has been written to SD_BUF

1. Flag register clear
   First, clear the bits in the flag register. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write H’0000 0019 to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting of the INFO2 bit (access end) bit in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write H’0000 FFFB to SD_INFO1_MASK to enable the access end interrupt. In addition, write H’0000 7D80 to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
   Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

40.4.14 Example of SD_CMD Register Setting

Table 40.8 lists the example of SD_CMD (SD interface) register setting.

Table 40.8 Example of SD_CMD Register Setting (SD) (1/2)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	H’0000 0000	
	CMD2	H’0000 0002	
	CMD3	H’0000 0003	
	CMD4	H’0000 0004	
	CMD5	H’0000 0705 or H’0000 0005	
	CMD6	H’0000 1C06 or H’0000 0006	
	CMD7	H’0000 0007	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	H’0000 0408 or H’0000 0008	
	CMD9	H’0000 0009	
	CMD10	H’0000 000A	
	CMD11	H’0000 040B or H’0000 000B	
	CMD12	H’0000 000C	
	CMD13	H’0000 000D	
	CMD15	H’0000 000F	
	CMD16	H’0000 0010	
	CMD17	H’0000 0011	
	CMD18	H’0000 0012	With auto CMD12 enabled (other than in SDR104 mode)
		H’0000 7C12	With auto CMD12 disabled (only in SDR104 mode)
	CMD19	H’0000 1C13 or H’0000 0013	
	CMD20	H’0000 0514 or H’0000 0014	
	CMD23	H’0000 0417 or H’0000 0017	
	CMD24	H’0000 0018	
	CMD25	H’0000 0019	With auto CMD12 enabled (other than in SDR104 mode)
		H’0000 6C19	With auto CMD12 disabled (only in SDR104 mode)
	CMD27	H’0000 001B	
	CMD28	H’0000 001C	
	CMD29	H’0000 001D	
	CMD30	H’0000 001E	
	CMD32	H’0000 0020	
	CMD33	H’0000 0021	
	CMD38	H’0000 0026	
	CMD42	H’0000 002A	
	CMD48	H’0000 1C30	
	CMD49	H’0000 0C31	
	CMD52	H’0000 0434 or H’0000 0034	

Table 40.8 Example of SD_CMD Register Setting (SD) (2/2)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD53	H'0000 1C35	Single read
		H'0000 0C35	Single write
		H'0000 7C35	Multiple read
		H'0000 6C35	Multiple write
		H'0000 0035	The value on the left can be set irrespective of whether single or multi. However, the CF39 bit in SD_ARG must be set as follows. Read: 0 Write: 1
	CMD55	H'0000 0037	
	CMD56	H'0000 0038	
	CMD58	H'0000 7C3A	
	CMD59	H'0000 6C3B	
ACMD	ACMD6	H'0000 0046	
	ACMD13	H'0000 004D	
	ACMD22	H'0000 0056	
	ACMD23	H'0000 0057	
	ACMD41	H'0000 0069	
	ACMD42	H'0000 006A	
	ACMD51	H'0000 0073	

Table 40.9 lists the example of SD_CMD (MMC interface) register setting.

Table 40.9 Example of SD_CMD Register Setting (MMC)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	H'0000 0000	
	CMD1	H'0000 0701	
	CMD2	H'0000 0002	
	CMD3	H'0000 0003	
	CMD4	H'0000 0004	
	CMD5	H'0000 0505	
	CMD6	H'0000 0506	In the response busy state
		H'0000 0406	Not in the response busy state
	CMD7	H'0000 0007	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	H'0000 1C08	
	CMD9	H'0000 0009	
	CMD10	H'0000 000A	
	CMD12	H'0000 000C	
	CMD13	H'0000 000D	
	CMD14	H'0000 1C0E	SDIF_MODE must be set to H'0100 (with CRC16 disabled).
	CMD15	H'0000 000F	
	CMD16	H'0000 0010	
	CMD17	H'0000 0011	
	CMD18	H'0000 7C12	Pre-defined
	CMD19	H'0000 0C13	SDIF_MODE must be set to H'0100 (with CRC16 disabled).
	CMD21	H'0000 1C15	
	CMD23	H'0000 0017	
	CMD24	H'0000 0018	
	CMD25	H'0000 6C19	Pre-defined
	CMD26	H'0000 0C1A	
	CMD27	H'0000 001B	
	CMD28	H'0000 001C	
	CMD29	H'0000 001D	
	CMD30	H'0000 001E	
	CMD31	H'0000 1C1F	
	CMD35	H'0000 0423	
	CMD36	H'0000 0424	
	CMD38	H'0000 0026	
	CMD39	H'0000 0427	
	CMD40	H'0000 0428	
	CMD42	H'0000 002A	
	CMD49	H'0000 0C31	
	CMD53	H'0000 7C35	
	CMD54	H'0000 6C36	
	CMD55	H'0000 0037	
	CMD56	H'0000 0038	

40.5 Usage Notes

(1) SD_BUF Illegal Write Access

When writing data to SD_BUF0 after the single block write or multi block write command is issued, the data of the size specified by SD_SIZE must be written to.

If the data of the size which exceeds the size specified by SD_SIZE is written to, the ERR4 bit in SD_INFO2 is set to 1. In addition, the data written to SD_BUF0 may not be transmitted and it causes the SCLKDIVEN bit in SD_INFO2 to hold the value of 0. In such cases, clearing the SDRST bit in SOFT_RST to 0 and then restoring its value to 1 clears the SCLKDIVEN bit to 1.

However, for the single byte (in the case of 16- or 32-bit access) or three bytes (in the case of 32-bit access) when the number of bytes setting in SD_SIZE is odd, or the fraction of bytes when the number of bytes setting in SD_SIZE is even (in the case of 32-bit access), since the portion of dummy data writing is regarded as excess data and ignored, it is not within the scope of the above description (the fraction of bytes: the two bytes that are not in a four-byte unit).

(2) Block Number Limitation for Multiple Block Read

When performing a multiple block read of one or two blocks, depending on the timing with which the response register is read, the response value may not be read properly. This must be avoided by either of the following countermeasures.

1) When receiving one or two blocks of data, use single block reading.
2) Read the response to CMD18 from SD_RSP54.

<Mechanism of incorrect reading>

Figure 40.28 shows the processing flows of SD/MMC host interface (hardware) operation and software operation when a multiple block read is performed on two blocks. As shown in the incorrect operation of **Figure 40.28**, when an interrupt is generated on reception of the CMD18 response and the timing with which the SD card response register (SD_RSP10) is read by the interrupt is delayed, the data during the CMD12 response reception or the CMD12 response may be read. In the case of a multiple block read of three or more blocks, CMD12 is not issued until the block of data has been read, so this problem does not arise. Furthermore, in the case of a multiple block write, since the CMD25 response is read before the block of data is sent, the problem does not arise.

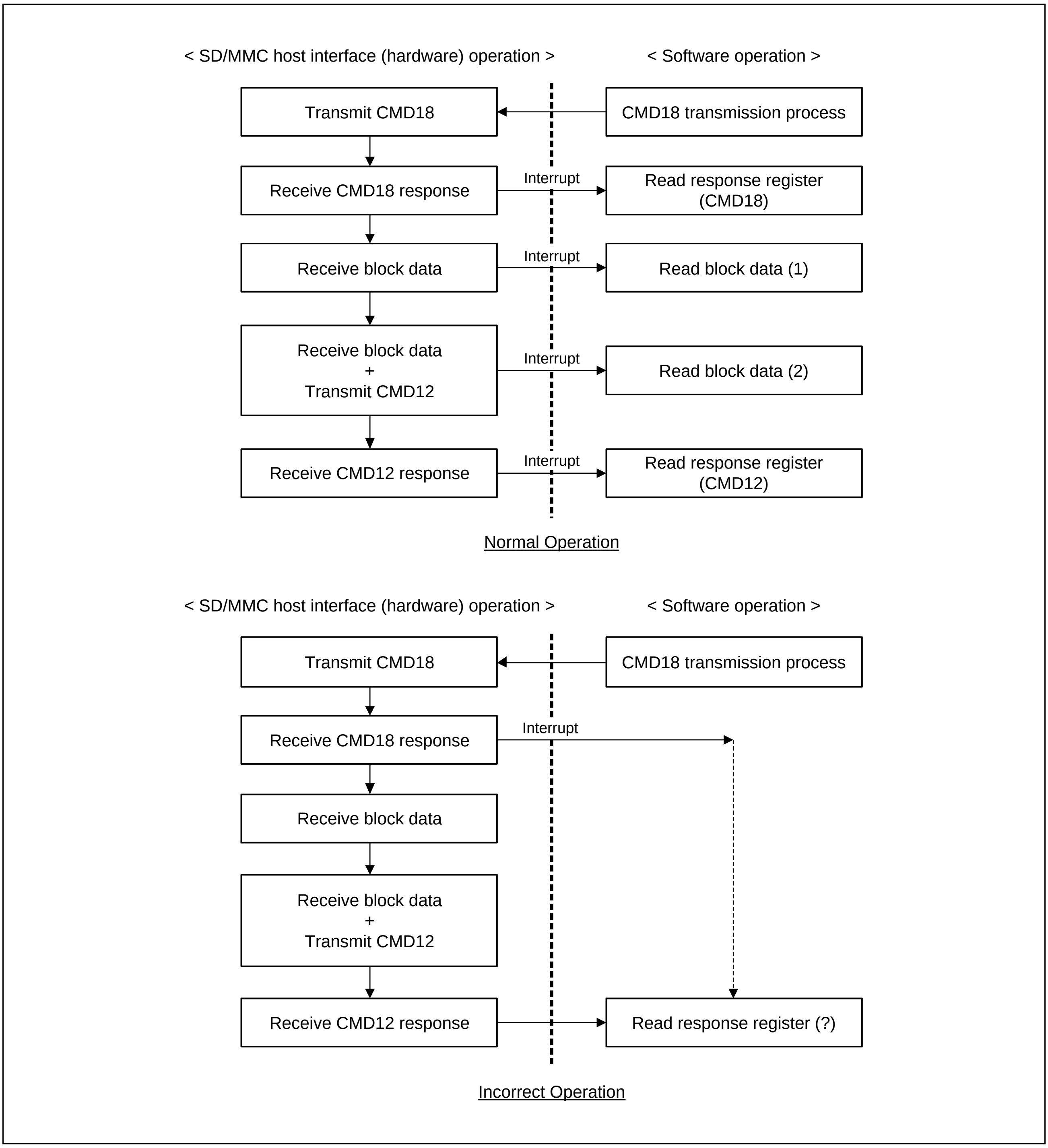


Figure 40.28 Flowcharts for Multiple Block Read Operation (Two Blocks)

(3) Automatic Control of SDCLK Output

In the SD Card standard, 74 cycles of SDCLK must be output before initialization of the card. For this reason, use automatic control of SDCLK output after 74 cycles of SDCLK have been output. Furthermore, if automatic control of SDCLK output was in use, SDCLK output is stopped on completion of the sequence for a communications error or timeout. Thus, in cases where state transitions within the SD card are necessary and so on after completion of the sequence, release automatic control of SDCLK output and restart supply of SDCLK to the SD card.

(4) Control of the C52PUB Setting for Multiple Block Write

If the C52PUB bit in SDIO_MODE is set to 1 during a sequence of multiple block write due to CMD53, CMD52 is not issued until SD_BUF becomes empty. For this reason, set the C52PUB bit after suspending writing to SD_BUF by following the appropriate procedure below.

- When DMA transfer is not in use

1. Before setting the C52PUB bit, suspend writing to SD_BUF by making the setting in SD_INFO2 to disable BWE interrupts.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by making the setting in SD_INFO2 to enable BWE interrupts.

- When DMA transfer is in use

1. Every time DMA transfer of the value set in SD_SIZE × n blocks (where n = 1, 2, ...) proceeds, suspend writing to SD_BUF by DMA transfer before the C52PUB bit is set.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by DMA transfer.

(5) Notes on SD_CLK_CTRL Register Settings

When the SCLKDIVEN bit in SD_INFO2 is 0, SD_CLK_CTRL cannot be written to. Before writing to SD_CLK_CTRL, be sure to check that the SCLKDIVEN bit in SD_INFO2 is 1.

(6) Restrictions on specifications

1. The SDIO suspend/resume is not supported.
2. The SPI bus is not supported.
3. The shared bus and 8-bit SD bus for embedded SDIO are not supported.
4. The stream transfer for MMC cards is not supported.

(7) STP bit setting during multiple block read

During execution of multiple block read with automatic CMD12 execution by setting the SEC bit in SD_STOP to 1, even if the STP bit in SD_STOP is set to 1 to forcibly stop the execution, the command sequence may not stop depending on the timing of setting the STP bit.

To avoid this, when setting the STP bit in SD_STOP to 1 during multiple block transfer, clear the SEC bit in SD_STOP to 0 at the same time. (Even when the SCLKDIVEN bit in SD_INFO2 is 0, change the SEC bit from 1 to 0.)

When the command sequence has not stopped because the SEC bit was not cleared to 0, the command sequence can be stopped by clearing the SDRST bit in SOFT_RST to 0.

When forcibly terminating the CMD53 multiple block transfer through the IOABT bit in SDIO_MODE, be sure to leave the SEC bit in SD_STOP as 1.

40.6 Sampling Clock Controller (SCC)

40.6.1 Features

This module controls a sampling clock (hereafter referred to as the SCC sampling clock) that is used for SD UHS-I/SDR104 and MMC HS200. When this module is used with the LSI, SD UHS-I/SDR104 and MMC HS200 can be supported.

40.6.2 SCC Block Diagram

Figure 40.29 shows a block diagram of the sampling clock controller.

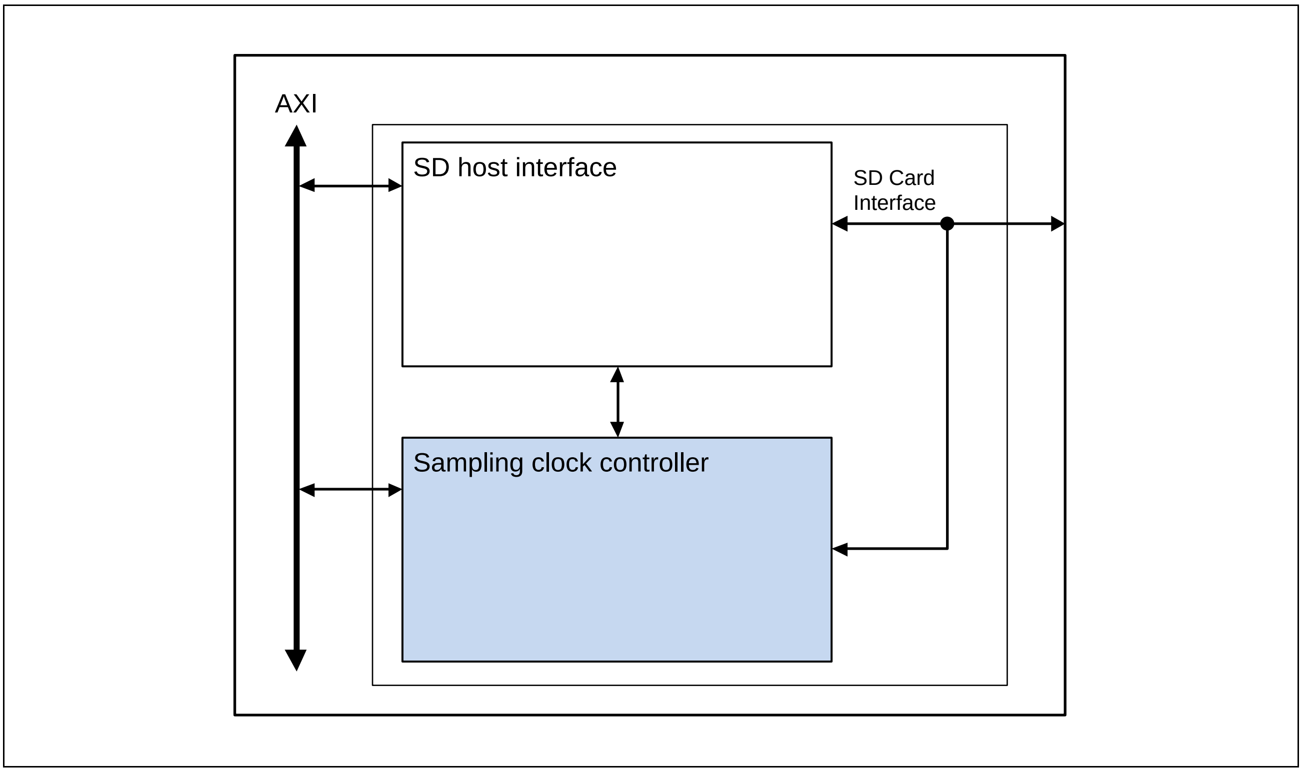


Figure 40.29 Block Diagram of the Sampling Clock Controller

40.6.3 SCC Register Configuration

Table 40.10 shows the SCC registers. Regarding the base address of registers as follows, refter to **Section 40.1.3**.

Channel 0: H’0_11C0_1000 (Cortex-A55)

Channel 0: H’41C0_1000 (Cortex-M33 Non-Secure)
H’51C0_1000 (Cortex-M33 Secure)

Channel 1: H’0_11C1_1000 (Cortex-A55)

Channel 1: H’41C1_1000 (Cortex-M33 Non-Secure)
H’51C1_1000 (Cortex-M33 Secure)

Table 40.10 Register Configuration

Name	Abbreviation	Address [9:0]	Access Size
Initial setting register	SCC_DTCNTL	H’1000	32/64
Sampling clock position setting register	SCC_TAPSET	H’1008	32/64
Hardware adjustment register 1	SCC_DT2FF	H’1010	32/64
Sampling clock selection register	SCC_CKSEL	H’1018	32/64
Sampling clock position correction register	SCC_RVSCNTL	H’1020	32/64
Sampling clock position correction request register	SCC_RVSREQ	H’1028	32/64
Sampling data comparison register	SCC_SMPCMP	H’1030	32/64
Hardware adjustment register 2	SCC_TMPPORT	H’1038	32/64

40.7 SCC Register Descriptions

40.7.1 Initial Setting Register (SCC_DTCNTL)

Bit	Bit Name	Initial Value	R/W	Description
63 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
23 to 16	TAPNUM7 to TAPNUM0	H’08	R/W	When bits DIV7 to DIV0 in the SD_CLK_CTRL register are H’FF (1: 1 mode), set these bits to H’08.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	TAPEN	0	R/W	SCC Sampling Clock Operation Enable 0: SCC sampling clock operation is disabled. 1: SCC sampling clock operation is enabled.

40.7.2 Sampling Clock Position Setting Register (SCC_TAPSET)

Bit	Bit Name	Initial Value	R/W	Description
63 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
7 to 0	TAPSET7 to TAPSET0	H'00	R/W	SCC Sampling Clock Position ● Set the tuning result in the range from 0 to TAPNUM-1.

40.7.3 Sampling Clock Selection Register (SCC_CKSEL)

Bit	Bit Name	Initial Value	R/W	Description
63 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
0	DTSEL	0	R/W	Sampling Clock Selection 0: An SCC sampling clock is not used (for other than SDR104 and HS200). 1: An SCC sampling clock is used (for SDR104 or HS200). ● For SDR104 or HS200, set DTSEL to 1. DIV[7:0] in the SD_CLK_CTRL register to H’FF (1:1 mode). ● When this bit is switched, stop the SD clock output from the SD/MMC host interface (set SCLKEN in SD_CLK_CTRL to 0).

40.7.4 Sampling Clock Position Correction Register (SCC_RVSCNTL)

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 8	TAPSEL7 to TAPSEL0	H'00	R	SCC Sampling Clock Position Display ● Displays the SCC sampling clock position selected by hardware. ● After RVSEN has been set to 1, the value may differ from that of TAPSET.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
1	RVSW	0	R/W	This bit is read as 0. The write value should be 1.
0	RVSEN	0	R/W	SCC Sampling Clock Position Correction Enable 0: SCC sampling clock position correction is disabled. 1: SCC sampling clock position correction is enabled. ● When RVSEN is set to 1 after tuning has been performed, this module corrects the SCC sampling clock position each time of a command sequence of the SD/MMC host interface. ● However, when RVSERR is 1, this module does not correct the SCC sampling clock position. ● While tuning is being performed, set RVSEN to 0.

40.7.5 Sampling Clock Position Correction Request Register (SCC_RVSREQ)

Bit	Bit Name	Initial Value	R/W	Description
63 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
2	RVSERR	0	R/W	SCC Sampling Clock Position Correction Error 0: There is no correction error. 1: There is a correction error. ● If this bit is set to 1 after a command sequence, write 0 to this bit and perform tuning again. ● Ignore this bit while tuning is being performed. ● Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.
1	REQTAPUP	0	R/W	SCC Sampling Clock Position Positive Direction Correction Request 0: There is no correction request. 1: There is a correction request. ● If this bit is set to 1 after a command sequence, write 0 to this bit and rewrite TAPSET in the positive direction (when TAPSEL = TAPNUM-1, set 0 to TAPSET). ● When RVSEN is 1, this bit is disabled (this bit is not set to 1). ● Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.
0	REQTAPDWN	0	R/W	SCC Sampling Clock Position Negative Direction Correction Request 0: There is no correction request. 1: There is a correction request. ● If this bit is set to 1 after a command sequence, write 0 to this bit and rewrite TAPSET in the negative direction (when TAPSEL = 0, set TAPNUM-1 to TAPSET). ● When RVSEN is 1, this bit is disabled (this bit is not set to 1). ● Writing 1 to this bit is disabled and writing 0 to this bit is only enabled.

40.7.6 Hardware Adjustment Register 1 (SCC_DT2FF)

This register makes a setting that SD_DATA, which has been fetched by the sampling clock at each TAP position, is used in the appropriate timing.

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 8	DT2NE	H’03	R/W	Hardware Adjustment 2 This is a setting register for adjusting the timing inside the IP when using Tuning. When using Tuning, set H’02 to this bit, make the setting before setting 1 in SCC_CKSEL [0] .DTSEL.
7 to 0	DT2NS	H’00	R/W	Hardware Adjustment 1 This is a setting register for adjusting the timing inside the IP when using Tuning. When using Tuning, set H’07 to this bit., make the setting before setting 1 in SCC_CKSEL [0] .DTSEL.

40.7.7 Sampling data comparison register (SCC_SMPCMP)

Data comparison register indicates the result of the comparison of the sampling data. The subject of the comparison is the before and the behind TAP.

Bit	Bit Name	Initial Value	R/W	Description
63 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
24 to 16	CMPNGU	All 0	R/W	Comparison of sampling data with the previous TAP Clock. Bit 16-23 is the comparison result of data 0-7. Bit 24 is the comparison result of CMD. 0: Match 1: Mismatch < Clear conditions.> The start of the command sequence Write to SCC_TAPSET register
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
8 to 0	CMPNGD	All 0	R/W	Comparison of sampling data with the after TAP Clock. Bit 0-7 is the comparison result of data 0-7. Bit 8 is the comparison result of CMD. 0: Match 1: Mismatch < Clear conditions.> The start of the command sequence Write to SCC_TAPSET register

40.7.8 Hardware adjustment register 2 (SCC_TMPPORT)

This register makes a setting that SD_DATA, which has been fetched by the sampling clock at each TAP position, is used in the appropriate timing.

Bit	Bit Name	Initial Value	R/W	Description
63 to16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
15 to 0	TMPOUT	H’0000	R/W	Hardware adjustment 3 When using the delay tuning mechanism, set the following values. When TAPNUM = 8, set H’0000 to these bits. When operating with other than SDR104 / HS200, this register adjusts the clock delay of the Flip Flop that latches the received data from the outside of the chip. For this product, set H’0000 when transferring at 3.3V, and set H’0001 when transferring at 1.8V.

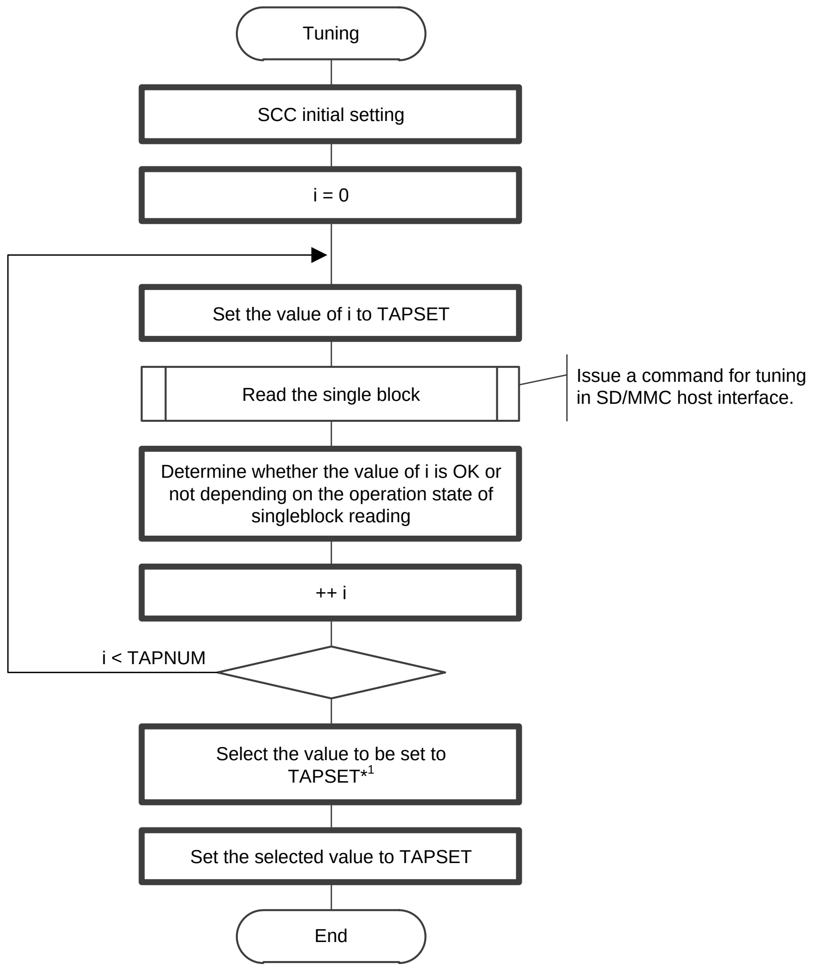
40.8 Usage Example of SCC

40.8.1 Tuning

SCC is tuned by using operation of single-block reading.

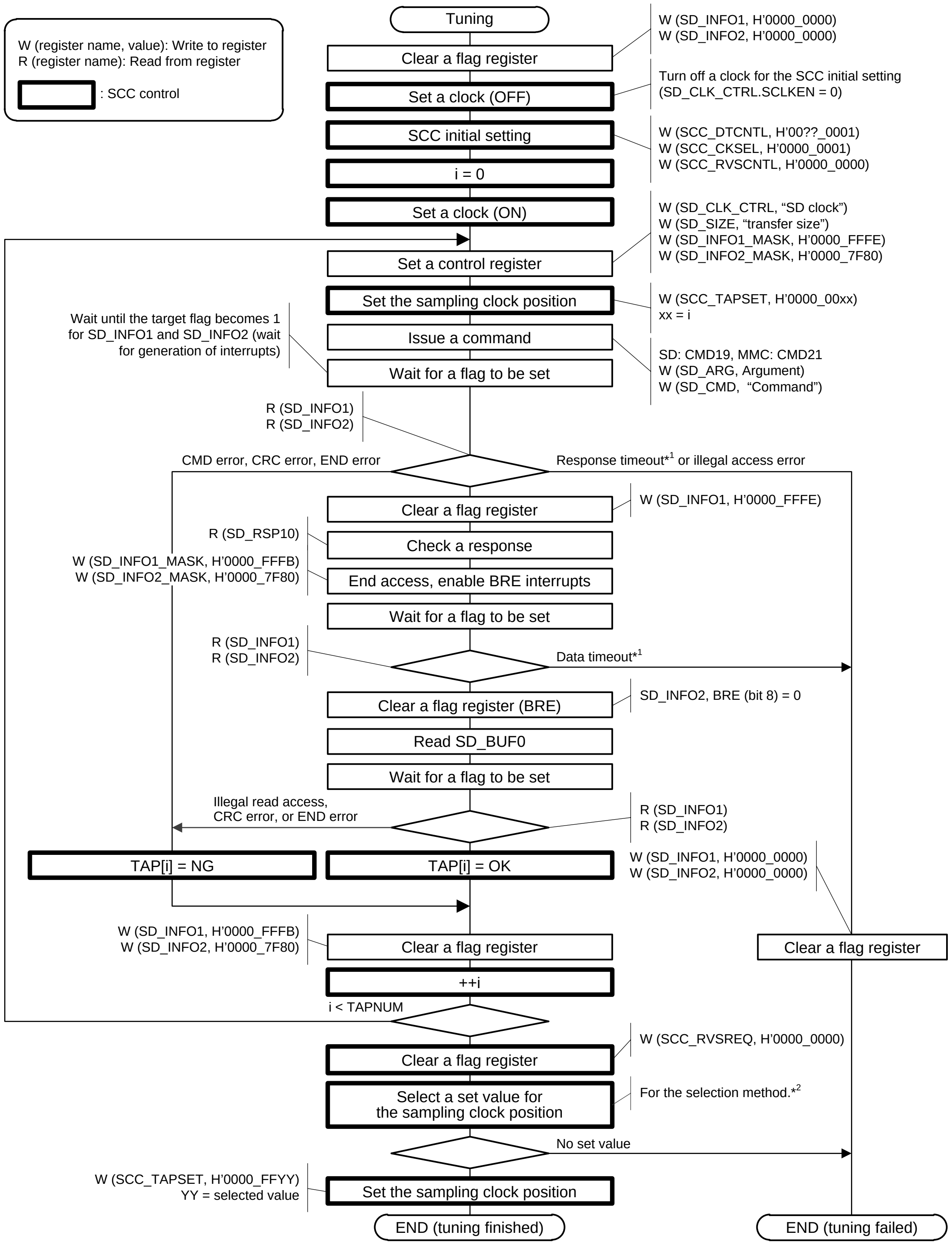
As shown in **Figure 40.30**, check whether the single-block read command normally ends when the sampling clock position is changed from 0 to TAPNUM-1 and save the result. After checking, confirm that there exists the range which has three or more continuous normal ends (OK). Then, the median value within the continuous range is determined as the final adjustment value.

Figure 40.31 and **Table 40.11** show the detailed tuning flow and the method how to select the sampling clock position (example when TAPNUM = 8).



Note 1. See the example shown in **Table 40.11**.

Figure 40.30 Example of Tuning Flow (Outline)



Note 1. If all of the TAP[i] has become "NG".

Note 2. See **Table 40.11**.

Figure 40.31 Example of Tuning Flow (Detailed)

Table 40.11 Example of the Method How to Select the Sampling Clock Position (when TAPNUM = 8)

Item	i	Case 1	Case 2	Case 3	Case 4	Case 5*[1]
TAP[i]	0	NG	OK	NG	OK	OK
	1	OK	OK	NG	NG	OK
	2	OK	NG	OK	NG	OK
	3	OK(←)	NG	OK	NG	OK
	4	OK	NG	NG	NG	OK
	5	OK	OK	NG	OK	OK
	6	NG	OK	NG	OK(←)	OK
Max. value→	7	NG	OK(←)	NG	OK	OK
	(0)	NG	OK	NG	OK	OK
	(1)	OK	OK	NG	NG	OK
	(2)	OK	NG	OK	NG	OK
	(3)	OK	NG	OK	NG	OK
	(4)	OK	NG	NG	NG	OK
	(5)	OK	OK	NG	OK	OK
	(6)	NG	OK	NG	OK	OK
	(7)	NG	OK	NG	OK	OK
Selected value		i = 3	i = 7	Fail	i = 6 or 7	i = 0 to 7

Remarks: (←): Example of selection, (x): repeated display of index x of TAP[x]

(a) The sampling clock position is selected by considering a margin in the range which has three or more continuous 'TAP[i] = OK'.

(b) The sampling clock position is repeated from 0 after the maximum value (TAPNUM-1). In case 2 above, that position is continued in the order of 5→6→7→0→1.

Note 1. If all of the TAP [i] is OK, the sampling clock position is selected by identifying the change point of data. Change point of the data can be found in the value of SCC_SMPCMP register. Usage example is **Section 40.8.3, Change point of the input data**.

40.8.2 Sampling Clock Position Correction after Tuning

After tuning, correction of the sampling clock position may be required when a command is issued.

There are manual and automatic correction methods. After a command sequence, if the CMD, CRC, END error or time out occurs or the correction error occurs, tuning will be performed again. The following shows examples of manual and automatic correction methods.

(1) Manual correction of the sampling clock position

Figure 40.32 shows the flow of manual correction of the sampling clock position. **Table 40.12** shows set values determined when correction is required (when TAPNUM = 8).

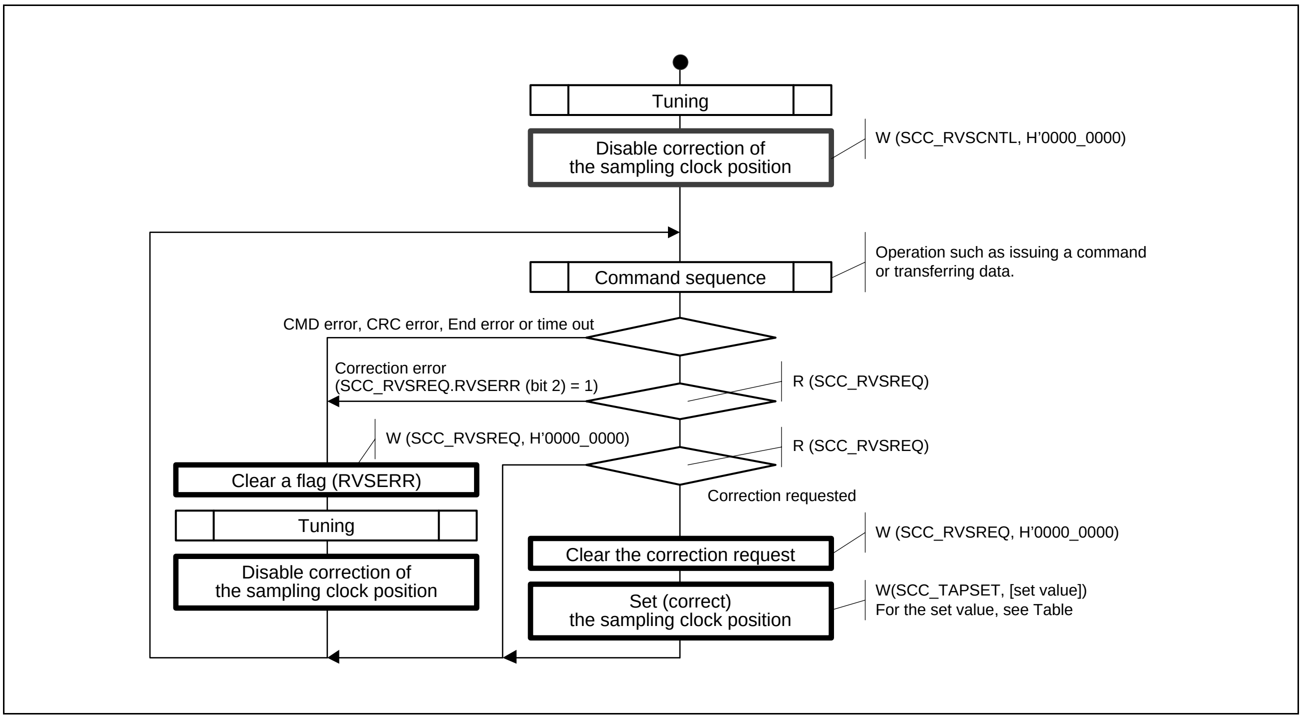


Figure 40.32 Flow of Manual Correction of the Sampling Clock Position (Example)

Table 40.12 Set Values for TAPSET when Correction is Required (when TAPNUM = 8)

No.	Current value of TAPSET	Value set to TAPSET when REQTAPUP = 1	Value set to TAPSET when REQTAPDWN = 1
1	0	1	7
2	1	2	0
3	2	3	1
4	3	4	2
5	4	5	3
6	5	6	4
7	6	7	5
8	7	0	6

Note: As is the case in the tuning selection method, the sampling clock position is 0 after the maximum value (TAPNUM-1).

(2) Automatic correction of the sampling clock position

Figure 40.33 shows the flow of automatic correction of the sampling clock position.

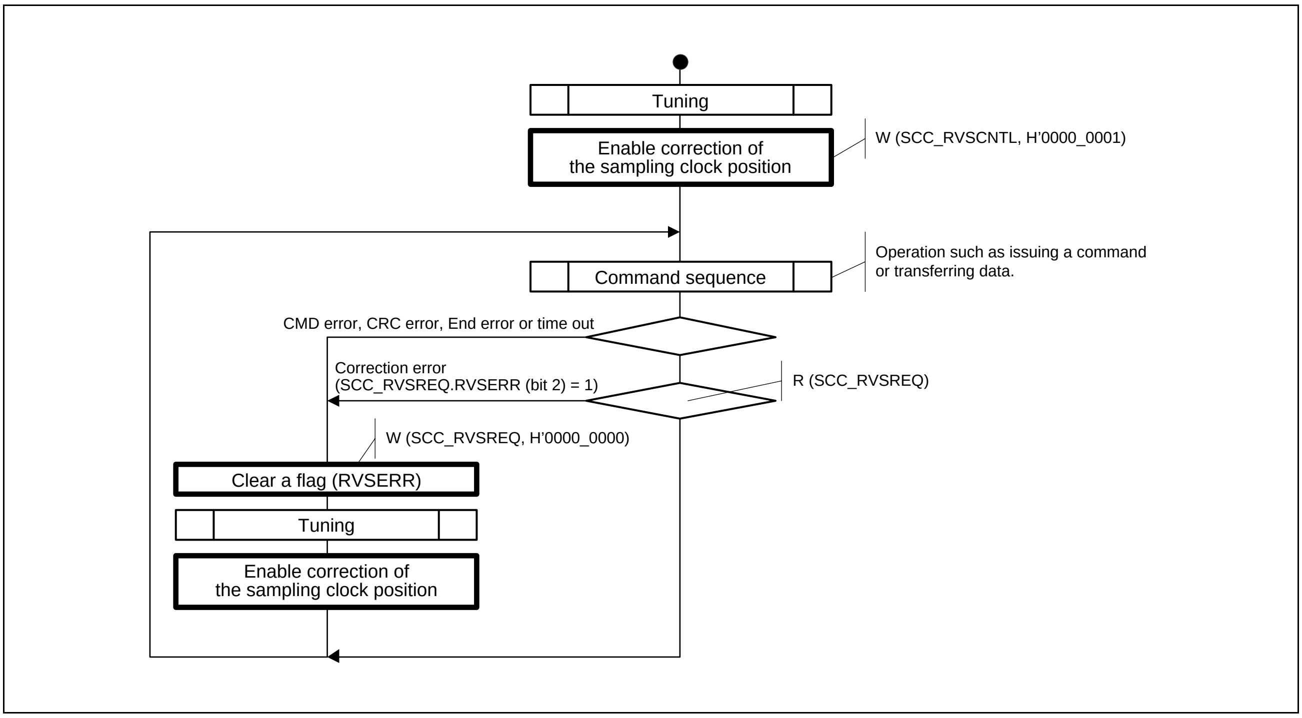
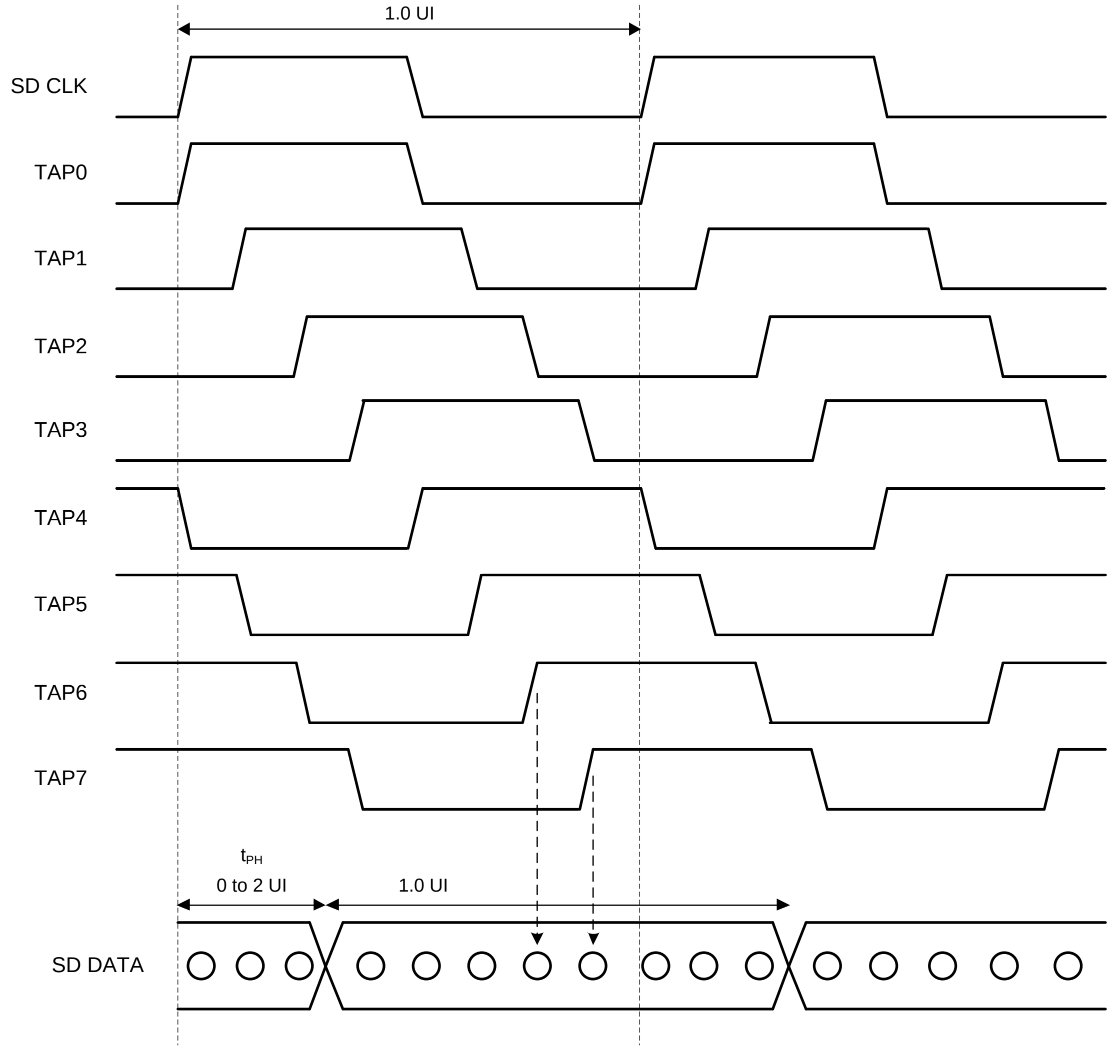


Figure 40.33 Flow of Automatic Correction of the Sampling Clock Position (Example)

40.8.3 Change point of the input data

Tuning captures the data by the TAP clock selected. However, also captures the data by the previous TAP clock and the behind the TAP clock at the same time. This result is reflected in the sampling data comparison register (SCC_SMPCMP). Point of mismatch before and after the selected TAP clock is the changing point of the data. In this example, it is desirable to set as TAP6 or TAP7.



Note: After recognizing the changing point of the input data, need to set the TAP clock.

Figure 40.34 Example, All taps is OK.

Figure 40.34 shows change point of data between TAP2 and TAP3. Change point of the data can be confirmed by sampling data comparison register (SCC_SMPCMP). When the tuning of TAP2 or TAP3, CMPNGU bit of sampling data comparison register indicates a mismatch. As the width of the input data is 1 (UI), select TAP6 or TAP7 which is the median of next TAP3 from TAP3.

41. General Purpose Input Output Port (GPIO)

This LSI can use up to 123 general-purpose I/O ports.

Each port of the general-purpose I/O port is multiplexed with the terminal of the peripheral function, and either the general-purpose I/O port function or the peripheral function can be selected by setting the register.

41.1 Features

41.1.1 General Purpose Input Output Port (GPIO)

General-purpose input/output ports (GPIOs) provide general-purpose ports that can be configured as inputs or outputs.

When set as an output, it can be written to an internal register to control the state driven by the output pin. When set as an input, you can read the state of the input by reading the state of the internal registers. You can also generate interrupts using the GPIO input pin.

- GPIO Output Control (Port Register [Pn])
- GPIO Input, Output Enable (Port Mode Register [PMn])
- GPIO Input Monitor (Port Input Register [PINn])
- Interrupt used the GPIO input (Interrupt Enable Control Register[ISEL])

Also, it able to control the following on some ports

- IO voltage mode control for each port
- Buffer drive ability control for each port
- Slew Rate Switching for each port
- Pull Up / Pull down Switching for each port
- Digital noise filter control for each port

41.1.2 Port Function Control

Switching of each port function can be controlled by the following register settings.

- GPIO function and peripheral function selection (Port Mode Control Register[PMCn])
- Selection of each peripheral function (Port Function Control Register[PFCm]) [Function0 to 5]

NOTE

Please refer to another file for the pin function for the peripheral functions that can be selected.
Please refer to above excel file for the“Pin function list” described in each chapter.

Do not select the same peripheral function for multiple different external terminals.

41.1.3 Special Purpose Port Function Control

The settings of the following specific ports can be controlled by registers.

- IO voltage mode control for each port
- Buffer drive ability control for each port
- Slew Rate Switching for each port
- Digital noise filter control for each port
- Ether MII / RGMII mode control

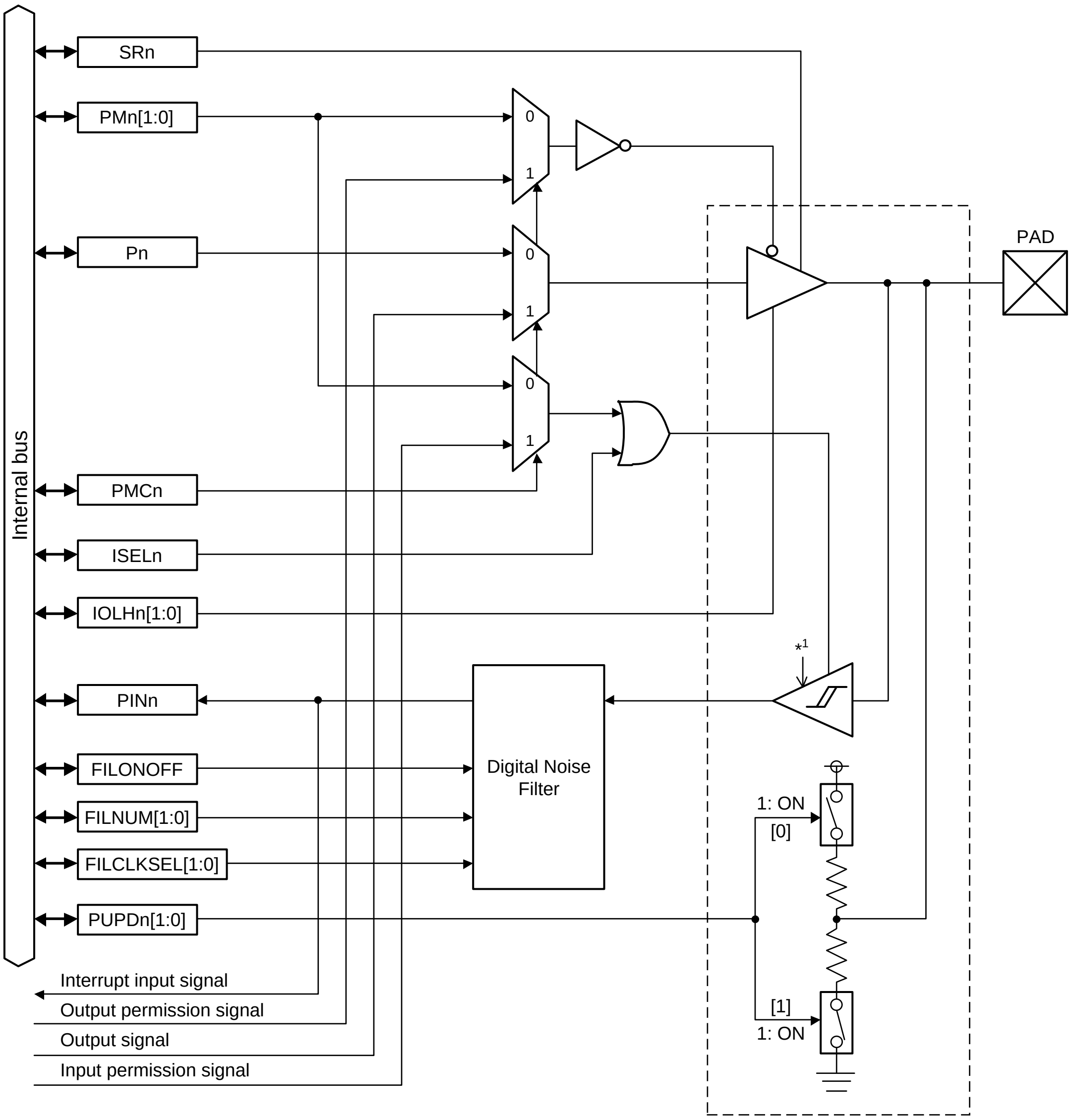
The specific ports shown below are targeted.

- Target Functions: Ether ch0/ch1, SDHI ch0/ch1, QSPI、RIIC
- Target Signals: NMI, TMS/SWDIO, TDO, WDTOVF_PERROUT#, AUDIO_CLK1, AUDIO_CLK2

41.1.4 General Purpose Input Output Port Configuration

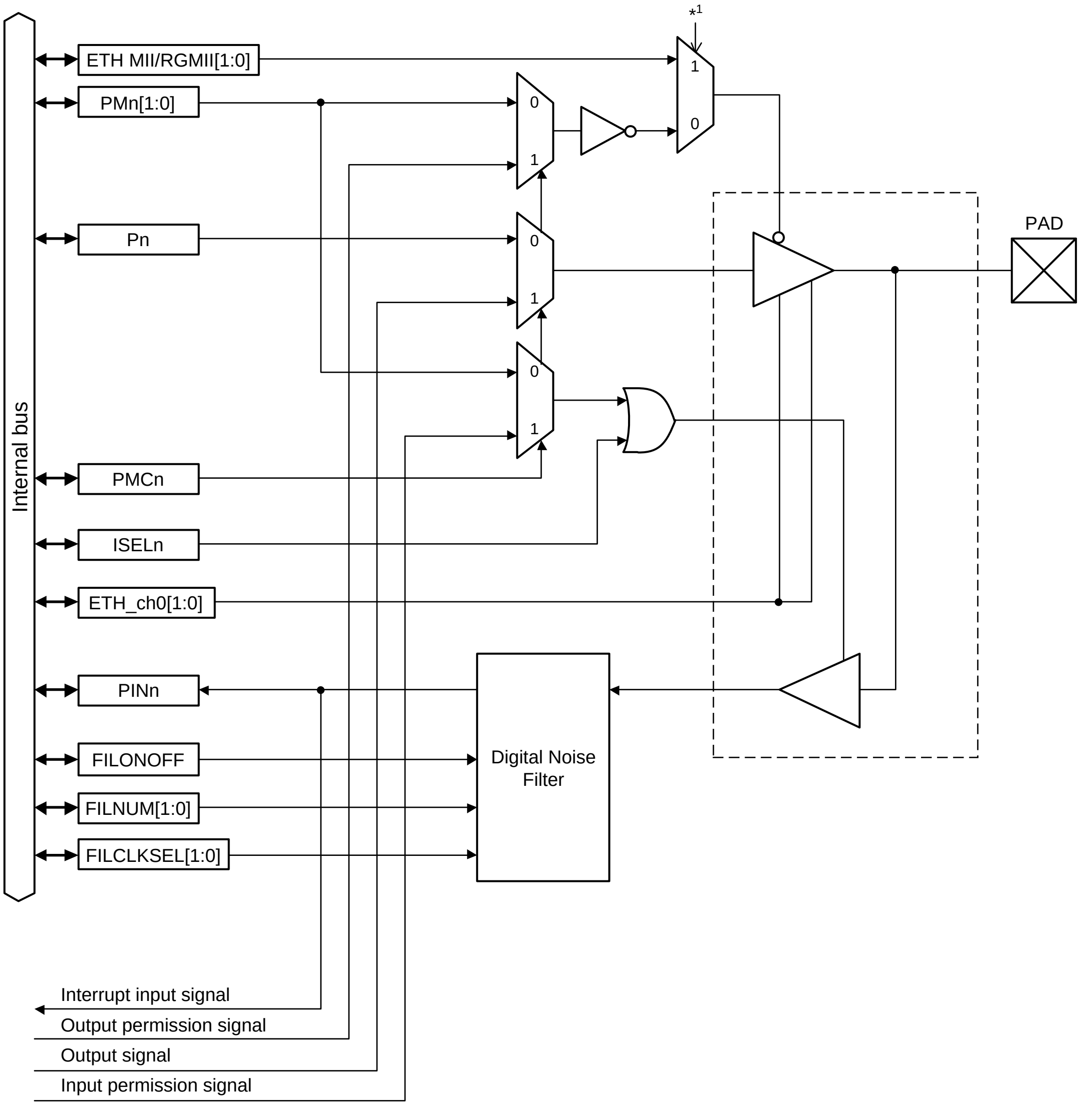
The configuration of the general-purpose I/O port is shown in **Figure 41.1** to **Figure 41.5**.

In addition, the configuration of the multiplexed Ether ch0 / ch1 is also included in the configuration diagram of the general-purpose I/O port.



Note 1. When RIIC * is selected as the peripheral module, SMT = 1 and the Schmitt trigger circuit is enabled.

Figure 41.1 P0_0 to P19_1, P38_0 to P48_4 (Multiplexed peripheral functions)



Note 1. When ETH0 is selected as the peripheral module, the ETH MII / RGMII register output is selected.

Figure 41.2 P20_0 (Multiplexed Ether ch0)

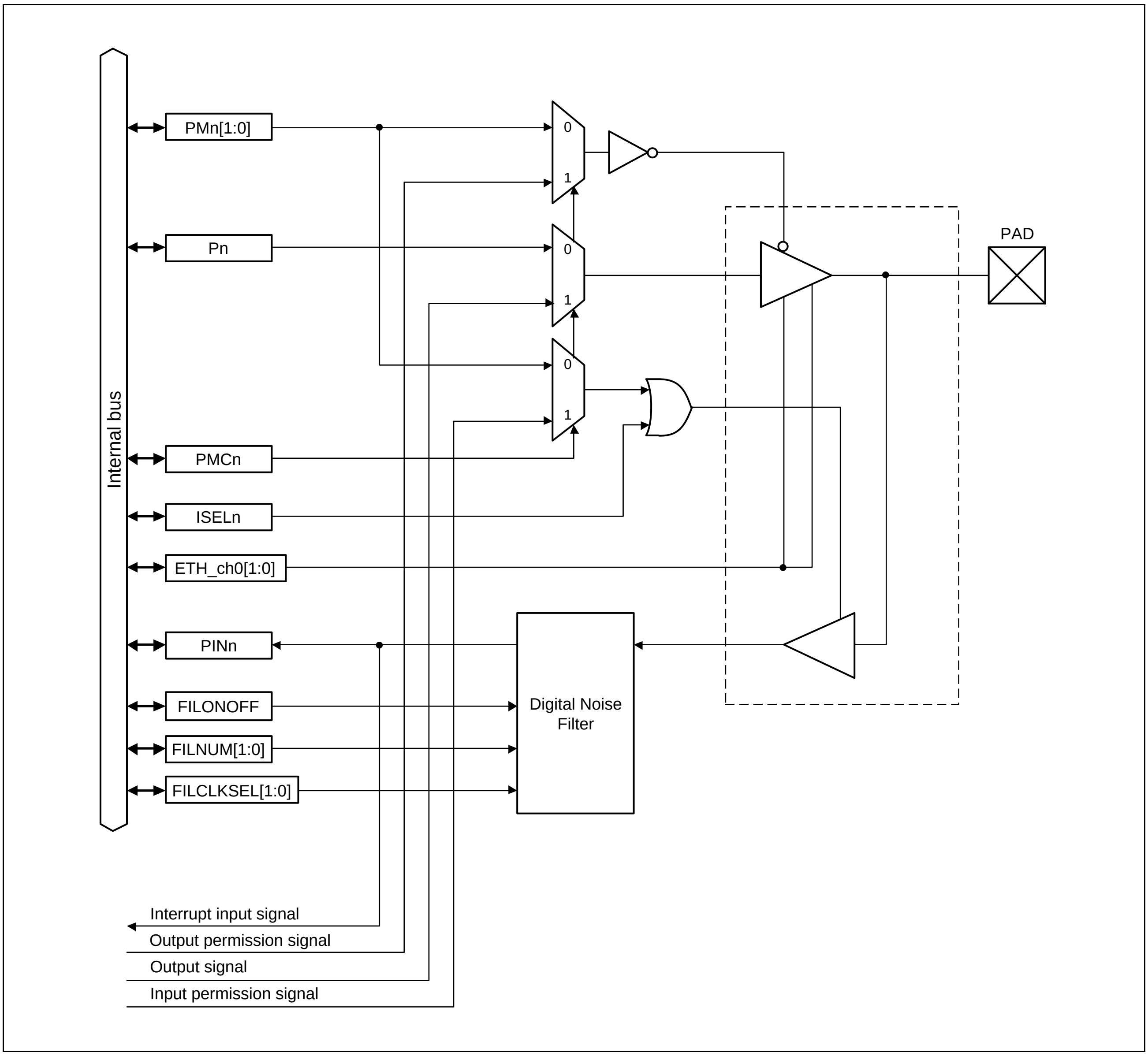
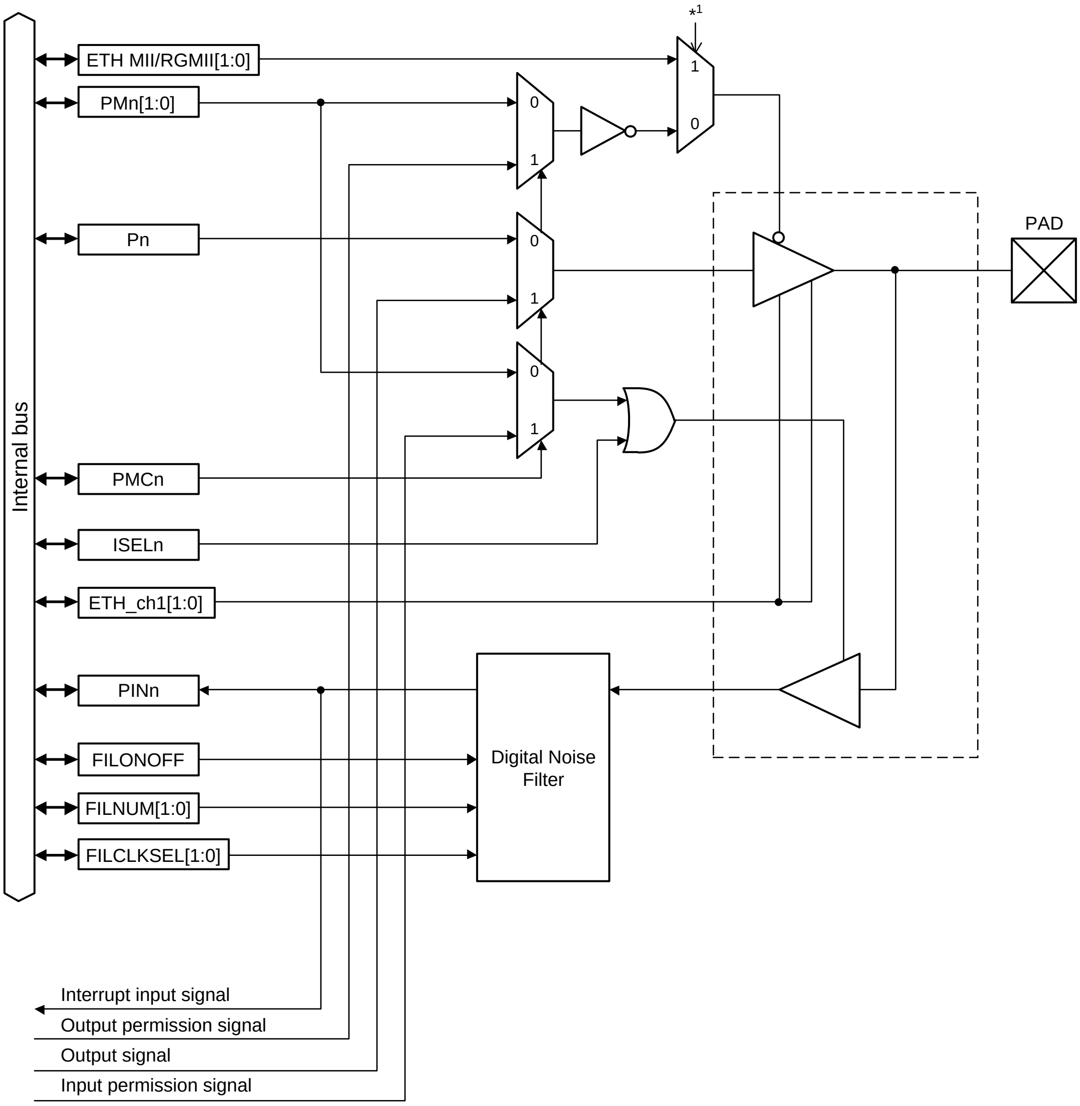


Figure 41.3 P20_1 to P28_1 (Multiplexed Ether ch0)



Note 1. When ETH1 is selected as the peripheral module, the ETH MII / RGMII register output is selected.

Figure 41.4 P29_0 (Multiplexed Ether ch1)

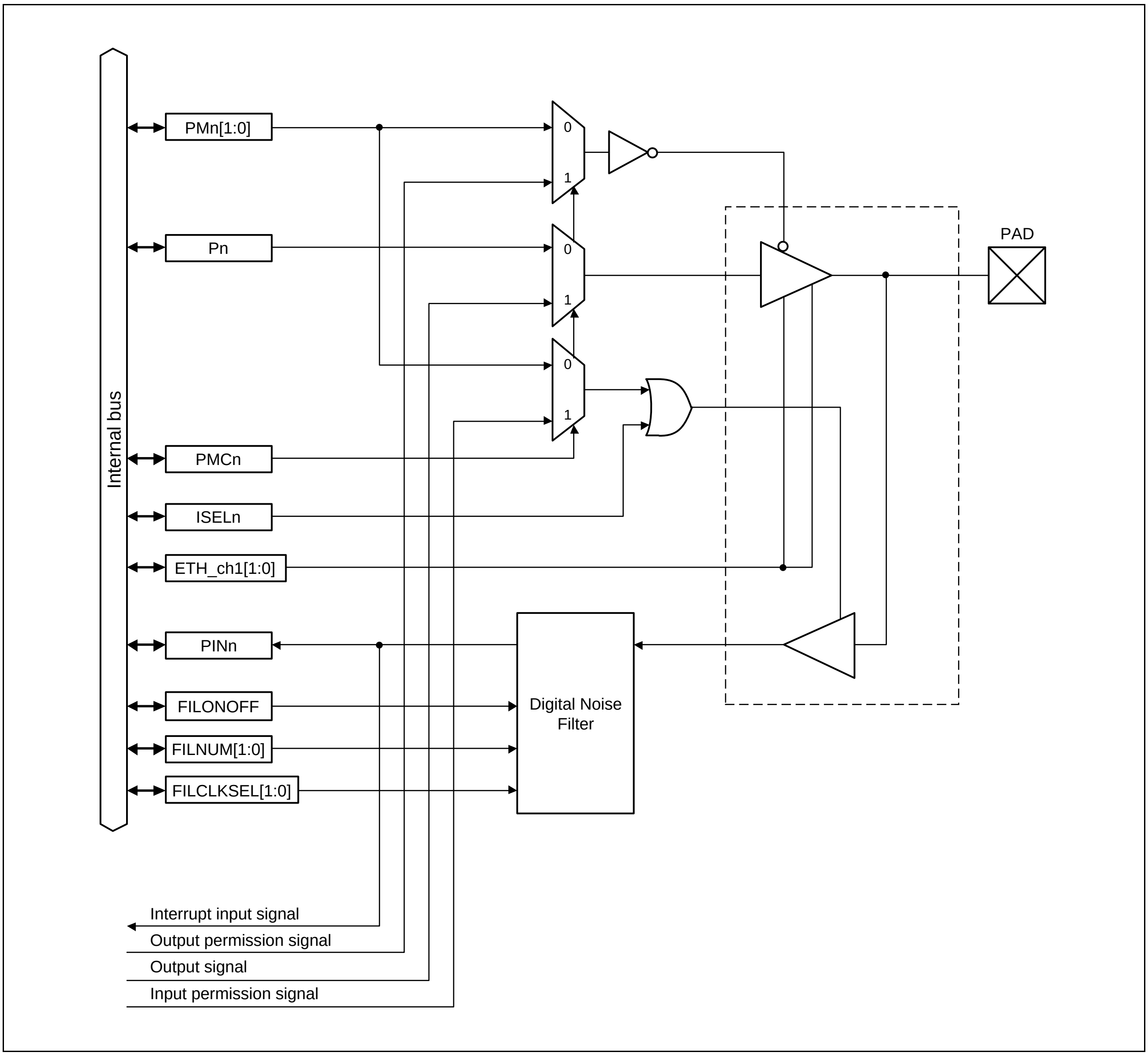
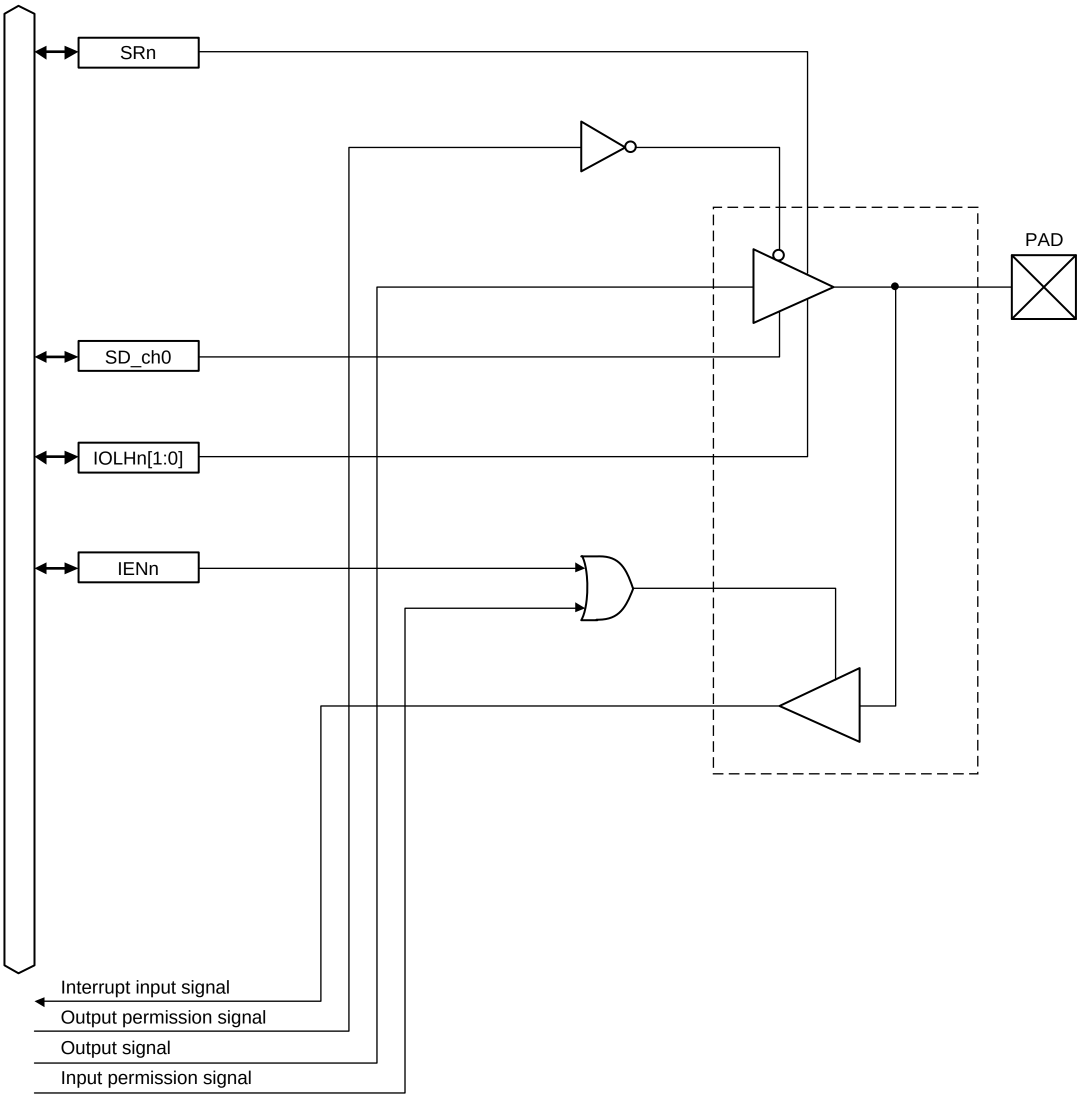


Figure 41.5 P29_1 to P37_2 (Multiplexed Ether ch1)

41.1.5 Special purpose port Configuration

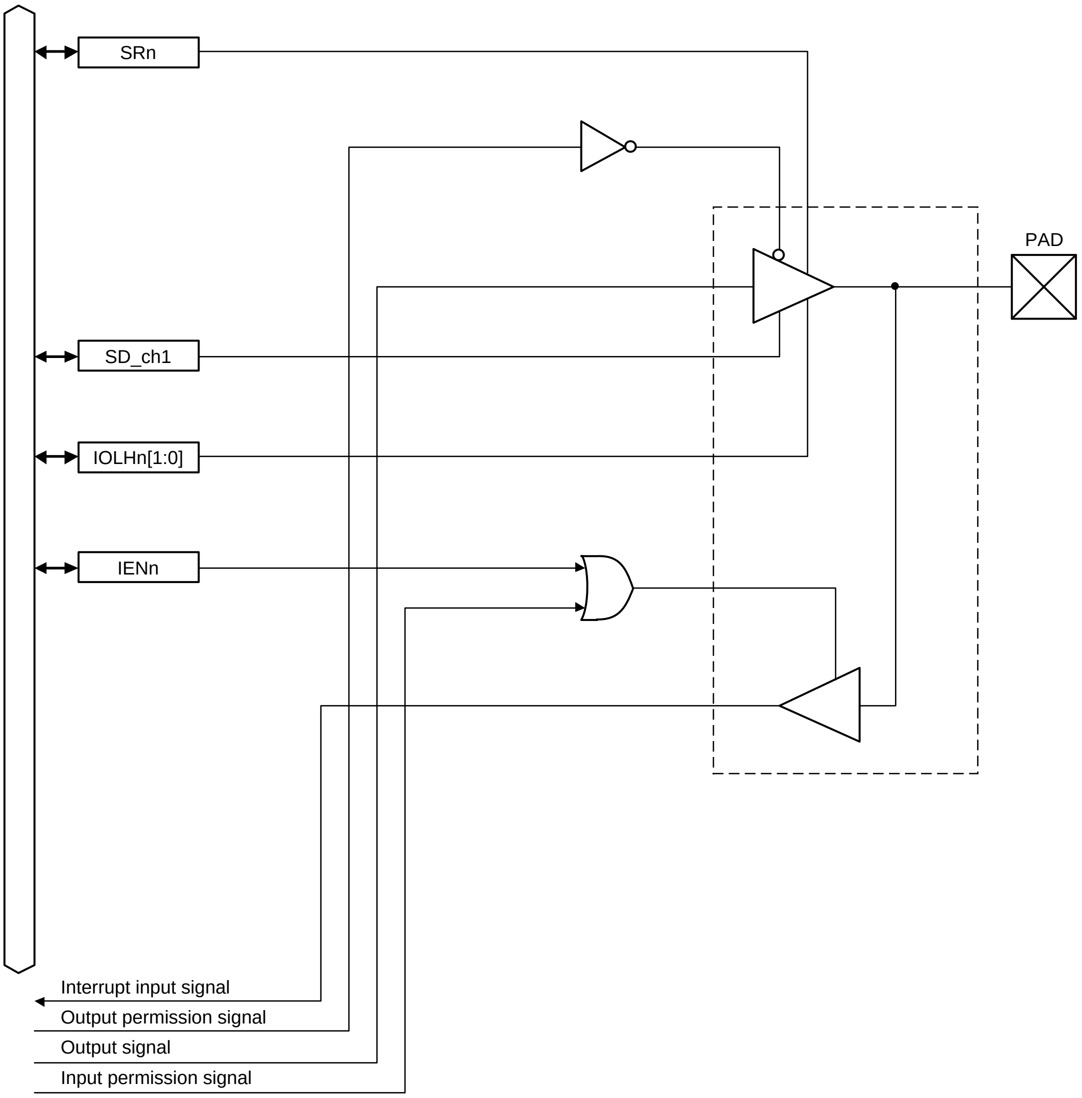
The functions and the signal of configurations shown below are shown in **Figure 41.6** to **Figure 41.13**.

- Target Functions: SDHI ch0, SDHI ch1, QSPI, RIIC
- Target Signals: NMI, TMS/SWDIO, TDO, WDTOVF_PERROUT#, AUDIO_CLK1, AUDIO_CLK2



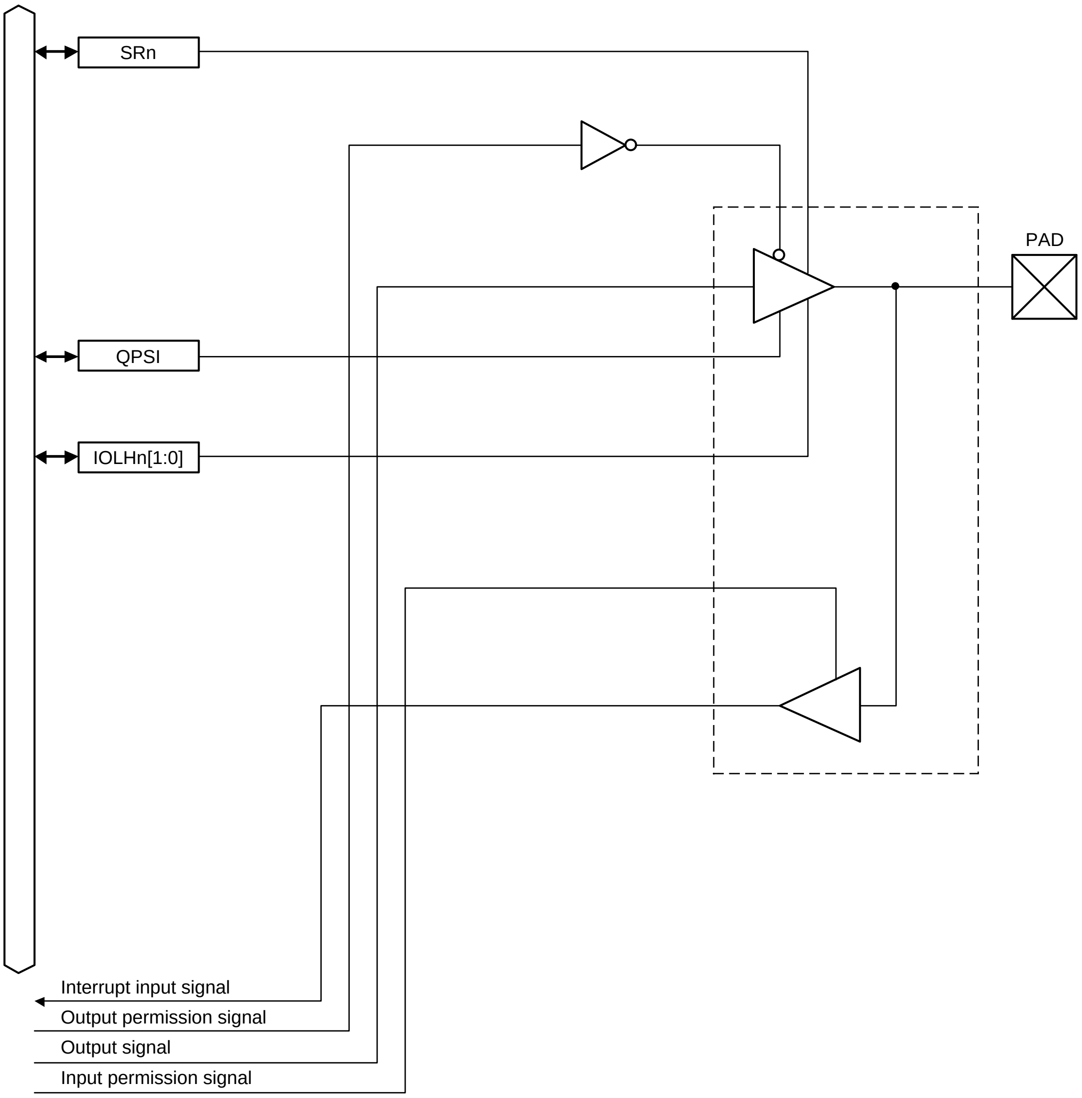
Note: The IENn and PUPDn registers do not exist at the SD0_CLK and SD0_RST# pins.

Figure 41.6 SDHI ch0 (SD0_CLK, SD0_CMD, SD0_RST#, SD0_DATA[7:0])



Note: The IENn and PUPDn registers do not exist at the SD1_CLK pin.

Figure 41.7 SDHI ch1 (SD1_CLK, SD1_CMD, SD1_DATA[3:0])



Note: The PUPDn register does not exist at the QSPI0_SPCLK, QSPI0_SSL, QSPI1_SPCLK, QSPI_RESET #, and QSPI_WP # pins.

Figure 41.8 QSPI (QSPI0_SPCLK, QSPI0_IO[3:0], QSPI0_SSL, QSPI1_SPCLK, QSPI1_IO[3:0], QSPI1_SSL, QSPI_RESET#, QSPI_WP#, QSPI_INT#)

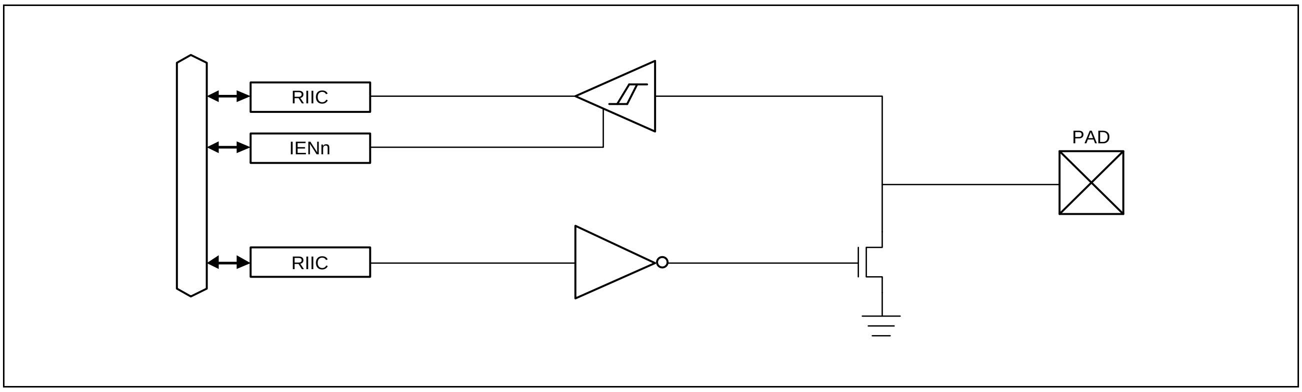


Figure 41.9 RIIC (RIIC0_SCL, RIIC0_SDA, RIIC1_SCL, RIIC1_SDA)

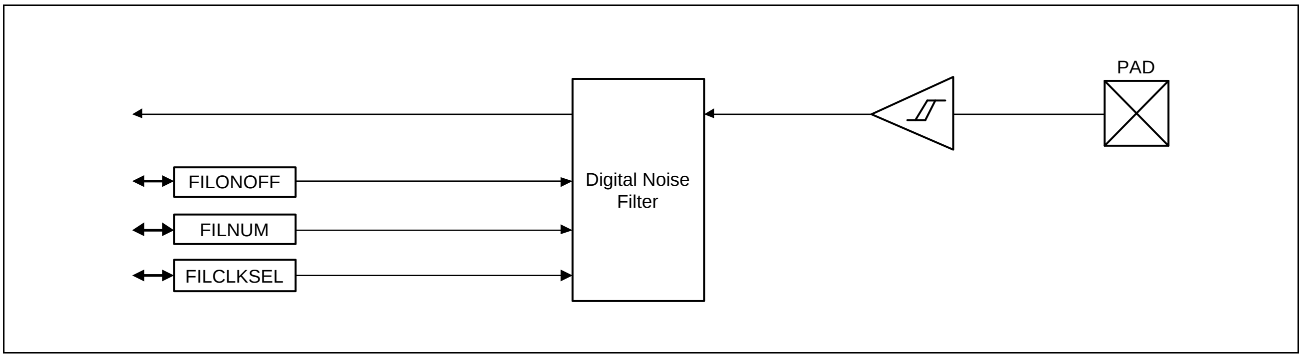


Figure 41.10 NMI

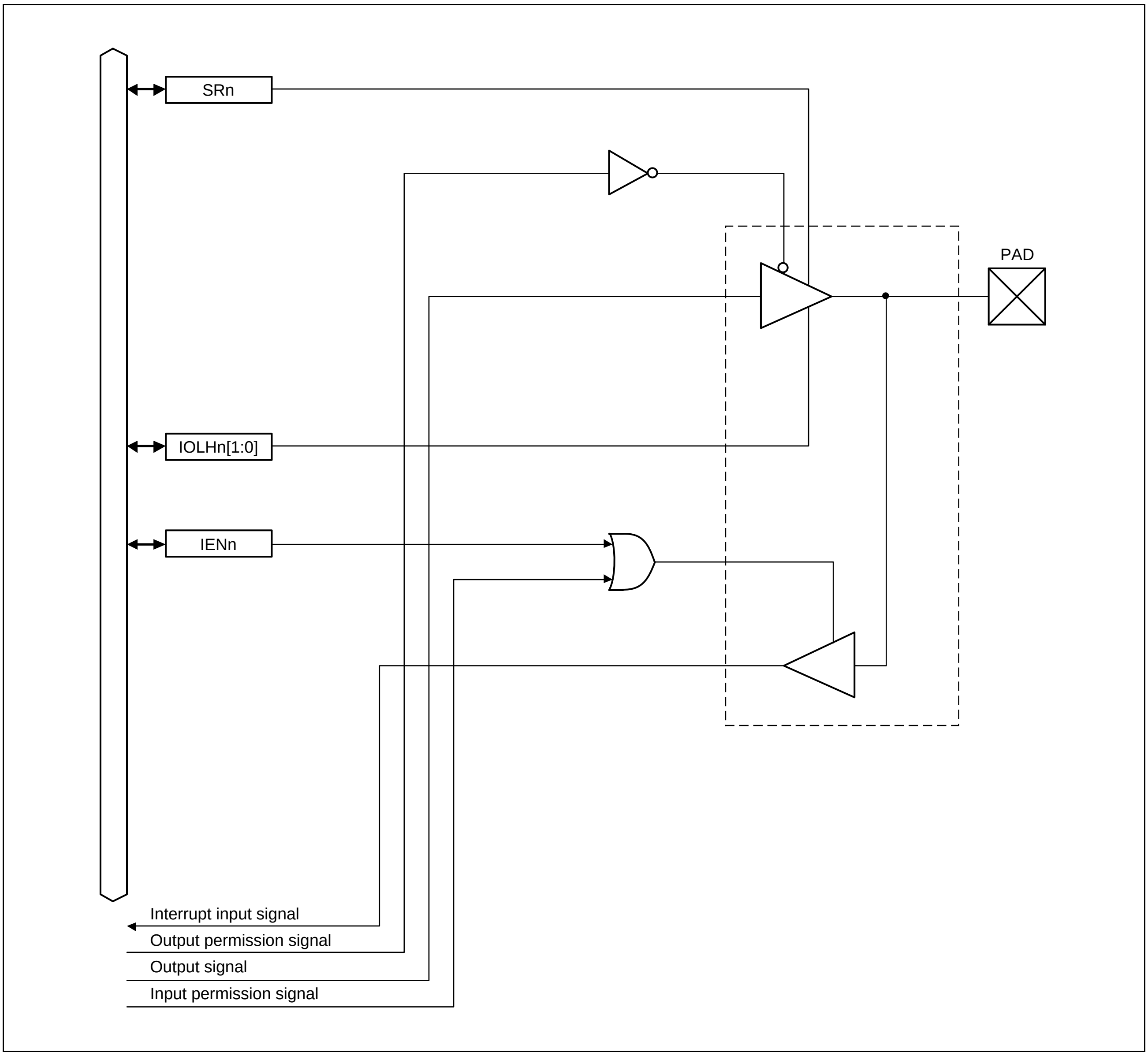


Figure 41.11 TMS/SWDIO

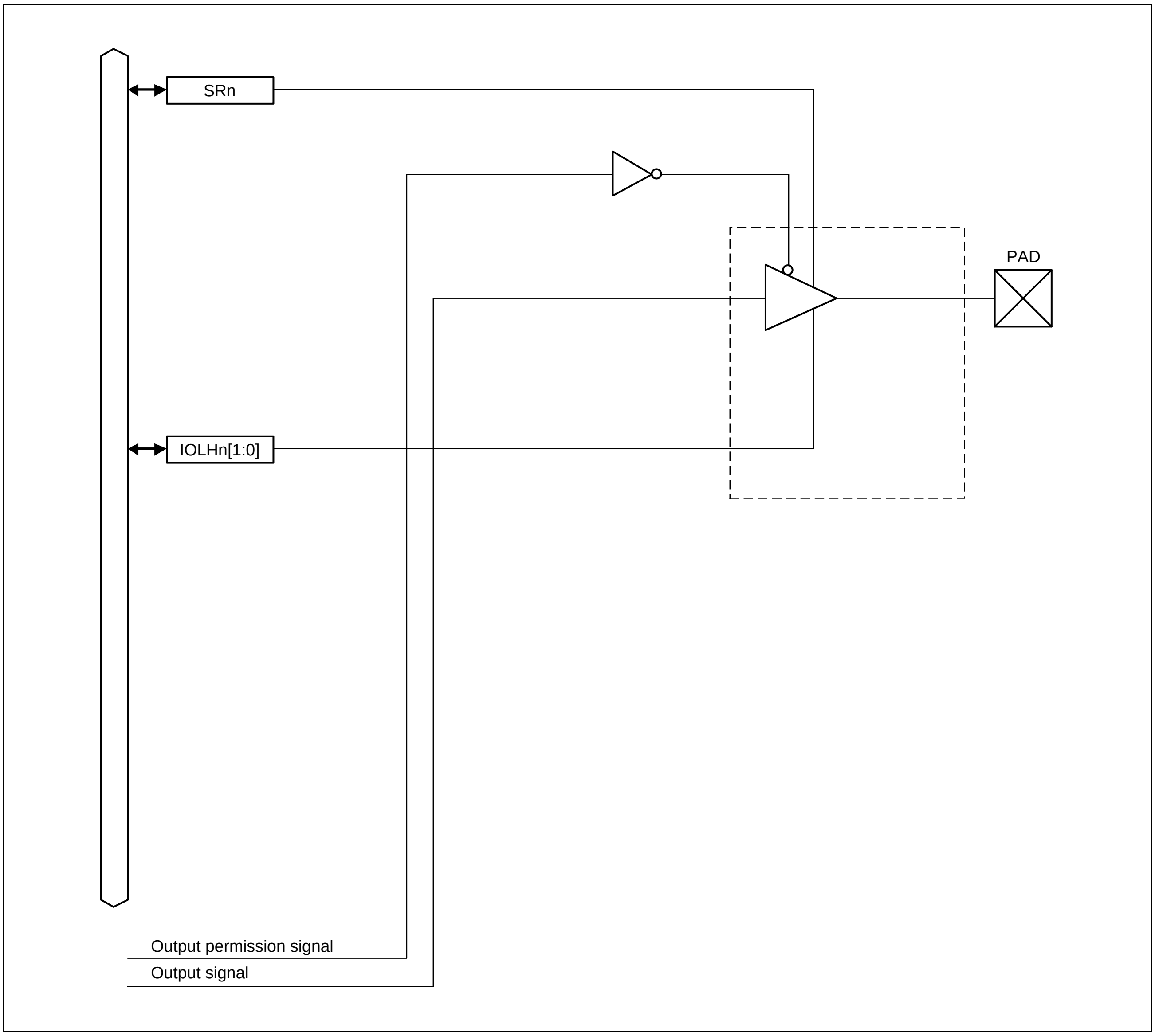


Figure 41.12 TDO, WDTOVF_PERROUT#

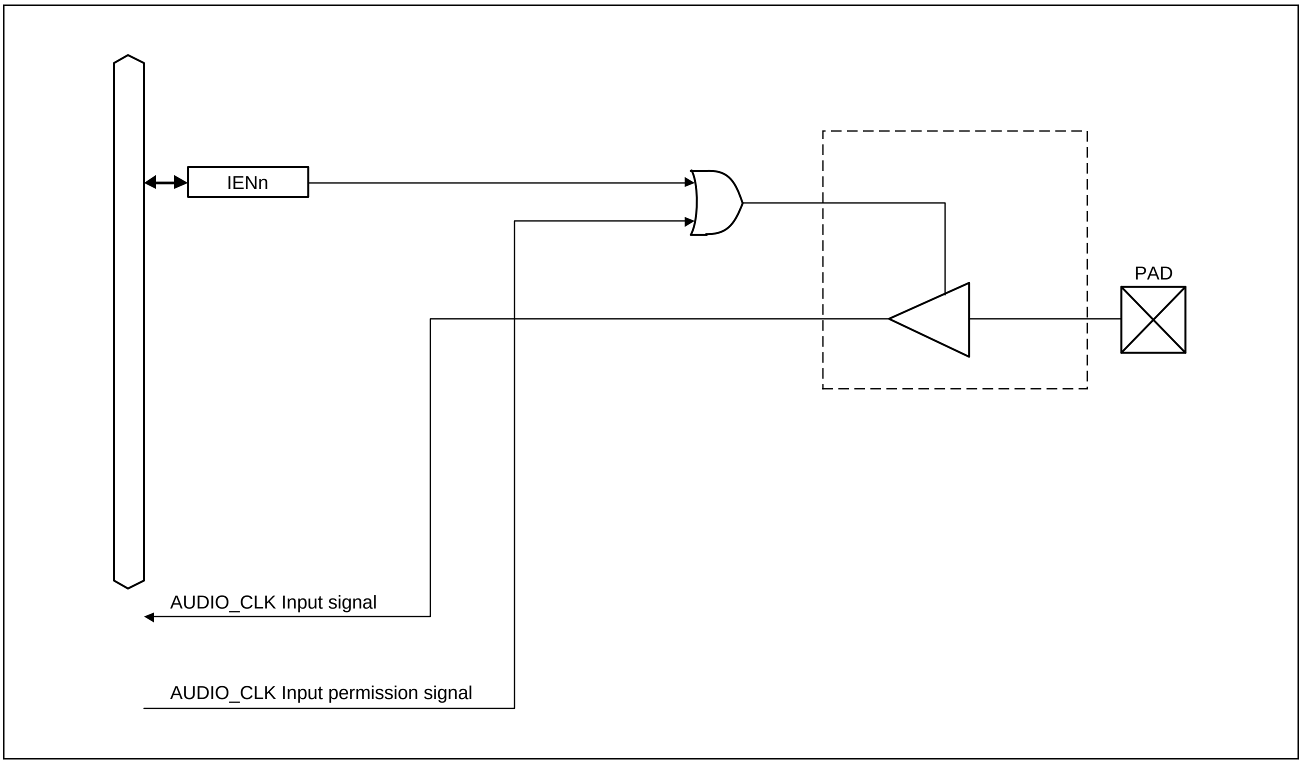


Figure 41.13 AUDIO_CLK1, AUDIO_CLK2

41.2 Register Configuration

The base address is shown below.

Base address: H’0_1103_0000 (Cortex-A55 Address Space)
Base address: H’4103_0000 (Cortex-M33 Address Space Non-Secure)
Base address: H’5103_0000 (Cortex-M33 Address Space Secure)

The register addresses shown in the following chapters will be offset addresses from the above base address.

NOTE

Access to areas for which Offset address is not defined in the register map is prohibited.

Please align the addresses according to the access size.

41.2.1 Port Register (Pn)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT REGISTER10	P10	R/W	H’00	H’0010	8
PORT REGISTER11	P11	R/W	H’00	H’0011	8
PORT REGISTER12	P12	R/W	H’00	H’0012	8
PORT REGISTER13	P13	R/W	H’00	H’0013	8
PORT REGISTER14	P14	R/W	H’00	H’0014	8
PORT REGISTER15	P15	R/W	H’00	H’0015	8
PORT REGISTER16	P16	R/W	H’00	H’0016	8
PORT REGISTER17	P17	R/W	H’00	H’0017	8
PORT REGISTER18	P18	R/W	H’00	H’0018	8
PORT REGISTER19	P19	R/W	H’00	H’0019	8
PORT REGISTER1A	P1A	R/W	H’00	H’001A	8
PORT REGISTER1B	P1B	R/W	H’00	H’001B	8
PORT REGISTER1C	P1C	R/W	H’00	H’001C	8
PORT REGISTER1D	P1D	R/W	H’00	H’001D	8
PORT REGISTER1E	P1E	R/W	H’00	H’001E	8
PORT REGISTER1F	P1F	R/W	H’00	H’001F	8
PORT REGISTER20	P20	R/W	H’00	H’0020	8
PORT REGISTER21	P21	R/W	H’00	H’0021	8
PORT REGISTER22	P22	R/W	H’00	H’0022	8
PORT REGISTER23	P23	R/W	H’00	H’0023	8
PORT REGISTER24	P24	R/W	H’00	H’0024	8
PORT REGISTER25	P25	R/W	H’00	H’0025	8
PORT REGISTER26	P26	R/W	H’00	H’0026	8
PORT REGISTER27	P27	R/W	H’00	H’0027	8
PORT REGISTER28	P28	R/W	H’00	H’0028	8
PORT REGISTER29	P29	R/W	H’00	H’0029	8
PORT REGISTER2A	P2A	R/W	H’00	H’002A	8
PORT REGISTER2B	P2B	R/W	H’00	H’002B	8
PORT REGISTER2C	P2C	R/W	H’00	H’002C	8

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT REGISTER2D	P2D	R/W	H’00	H’002D	8
PORT REGISTER2E	P2E	R/W	H’00	H’002E	8
PORT REGISTER2F	P2F	R/W	H’00	H’002F	8
PORT REGISTER30	P30	R/W	H’00	H’0030	8
PORT REGISTER31	P31	R/W	H’00	H’0031	8
PORT REGISTER32	P32	R/W	H’00	H’0032	8
PORT REGISTER33	P33	R/W	H’00	H’0033	8
PORT REGISTER34	P34	R/W	H’00	H’0034	8
PORT REGISTER35	P35	R/W	H’00	H’0035	8
PORT REGISTER36	P36	R/W	H’00	H’0036	8
PORT REGISTER37	P37	R/W	H’00	H’0037	8
PORT REGISTER38	P38	R/W	H’00	H’0038	8
PORT REGISTER39	P39	R/W	H’00	H’0039	8
PORT REGISTER3A	P3A	R/W	H’00	H’003A	8
PORT REGISTER3B	P3B	R/W	H’00	H’003B	8
PORT REGISTER3C	P3C	R/W	H’00	H’003C	8
PORT REGISTER3D	P3D	R/W	H’00	H’003D	8
PORT REGISTER3E	P3E	R/W	H’00	H’003E	8
PORT REGISTER3F	P3F	R/W	H’00	H’003F	8
PORT REGISTER40	P40	R/W	H’00	H’0040	8

41.2.2 Port Mode Register (PMn)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT MODE REGISTER10	PM10	R/W	H’0000	H’0120	8/16
PORT MODE REGISTER11	PM11	R/W	H’0000	H’0122	8/16
PORT MODE REGISTER12	PM12	R/W	H’0000	H’0124	8/16
PORT MODE REGISTER13	PM13	R/W	H’0000	H’0126	8/16
PORT MODE REGISTER14	PM14	R/W	H’0000	H’0128	8/16
PORT MODE REGISTER15	PM15	R/W	H’0000	H’012A	8/16
PORT MODE REGISTER16	PM16	R/W	H’0000	H’012C	8/16
PORT MODE REGISTER17	PM17	R/W	H’0000	H’012E	8/16
PORT MODE REGISTER18	PM18	R/W	H’0000	H’0130	8/16
PORT MODE REGISTER19	PM19	R/W	H’0000	H’0132	8/16
PORT MODE REGISTER1A	PM1A	R/W	H’0000	H’0134	8/16
PORT MODE REGISTER1B	PM1B	R/W	H’0000	H’0136	8/16
PORT MODE REGISTER1C	PM1C	R/W	H’0000	H’0138	8/16
PORT MODE REGISTER1D	PM1D	R/W	H’0000	H’013A	8/16
PORT MODE REGISTER1E	PM1E	R/W	H’0000	H’013C	8/16
PORT MODE REGISTER1F	PM1F	R/W	H’0000	H’013E	8/16
PORT MODE REGISTER20	PM20	R/W	H’0000	H’0140	8/16
PORT MODE REGISTER21	PM21	R/W	H’0000	H’0142	8/16
PORT MODE REGISTER22	PM22	R/W	H’0000	H’0144	8/16
PORT MODE REGISTER23	PM23	R/W	H’0000	H’0146	8/16
PORT MODE REGISTER24	PM24	R/W	H’0000	H’0148	8/16
PORT MODE REGISTER25	PM25	R/W	H’0000	H’014A	8/16

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT MODE REGISTER26	PM26	R/W	H’0000	H’014C	8/16
PORT MODE REGISTER27	PM27	R/W	H’0000	H’014E	8/16
PORT MODE REGISTER28	PM28	R/W	H’0000	H’0150	8/16
PORT MODE REGISTER29	PM29	R/W	H’0000	H’0152	8/16
PORT MODE REGISTER2A	PM2A	R/W	H’0000	H’0154	8/16
PORT MODE REGISTER2B	PM2B	R/W	H’0000	H’0156	8/16
PORT MODE REGISTER2C	PM2C	R/W	H’0000	H’0158	8/16
PORT MODE REGISTER2D	PM2D	R/W	H’0000	H’015A	8/16
PORT MODE REGISTER2E	PM2E	R/W	H’0000	H’015C	8/16
PORT MODE REGISTER2F	PM2F	R/W	H’0000	H’015E	8/16
PORT MODE REGISTER30	PM30	R/W	H’0000	H’0160	8/16
PORT MODE REGISTER31	PM31	R/W	H’0000	H’0162	8/16
PORT MODE REGISTER32	PM32	R/W	H’0000	H’0164	8/16
PORT MODE REGISTER33	PM33	R/W	H’0000	H’0166	8/16
PORT MODE REGISTER34	PM34	R/W	H’0000	H’0168	8/16
PORT MODE REGISTER35	PM35	R/W	H’0000	H’016A	8/16
PORT MODE REGISTER36	PM36	R/W	H’0000	H’016C	8/16
PORT MODE REGISTER37	PM37	R/W	H’0000	H’016E	8/16
PORT MODE REGISTER38	PM38	R/W	H’0000	H’0170	8/16
PORT MODE REGISTER39	PM39	R/W	H’0000	H’0172	8/16
PORT MODE REGISTER3A	PM3A	R/W	H’0000	H’0174	8/16
PORT MODE REGISTER3B	PM3B	R/W	H’0000	H’0176	8/16
PORT MODE REGISTER3C	PM3C	R/W	H’0000	H’0178	8/16
PORT MODE REGISTER3D	PM3D	R/W	H’0000	H’017A	8/16
PORT MODE REGISTER3E	PM3E	R/W	H’0000	H’017C	8/16
PORT MODE REGISTER3F	PM3F	R/W	H’0000	H’017E	8/16
PORT MODE REGISTER40	PM40	R/W	H’0000	H’0180	8/16

41.2.3 Port Mode Control Register (PMCn)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT MODE CONTROL REGISTER10	PMC10	R/W	H’00	H’0210	8
PORT MODE CONTROL REGISTER11	PMC11	R/W	H’00	H’0211	8
PORT MODE CONTROL REGISTER12	PMC12	R/W	H’00	H’0212	8
PORT MODE CONTROL REGISTER13	PMC13	R/W	H’00	H’0213	8
PORT MODE CONTROL REGISTER14	PMC14	R/W	H’00	H’0214	8
PORT MODE CONTROL REGISTER15	PMC15	R/W	H’00	H’0215	8
PORT MODE CONTROL REGISTER16	PMC16	R/W	H’00	H’0216	8
PORT MODE CONTROL REGISTER17	PMC17	R/W	H’00	H’0217	8
PORT MODE CONTROL REGISTER18	PMC18	R/W	H’00	H’0218	8
PORT MODE CONTROL REGISTER19	PMC19	R/W	H’00	H’0219	8
PORT MODE CONTROL REGISTER1A	PMC1A	R/W	H’00	H’021A	8
PORT MODE CONTROL REGISTER1B	PMC1B	R/W	H’00	H’021B	8
PORT MODE CONTROL REGISTER1C	PMC1C	R/W	H’00	H’021C	8
PORT MODE CONTROL REGISTER1D	PMC1D	R/W	H’00	H’021D	8
PORT MODE CONTROL REGISTER1E	PMC1E	R/W	H’00	H’021E	8
PORT MODE CONTROL REGISTER1F	PMC1F	R/W	H’00	H’021F	8
PORT MODE CONTROL REGISTER20	PMC20	R/W	H’00	H’0220	8
PORT MODE CONTROL REGISTER21	PMC21	R/W	H’00	H’0221	8
PORT MODE CONTROL REGISTER22	PMC22	R/W	H’00	H’0222	8
PORT MODE CONTROL REGISTER23	PMC23	R/W	H’00	H’0223	8
PORT MODE CONTROL REGISTER24	PMC24	R/W	H’00	H’0224	8
PORT MODE CONTROL REGISTER25	PMC25	R/W	H’00	H’0225	8
PORT MODE CONTROL REGISTER26	PMC26	R/W	H’00	H’0226	8
PORT MODE CONTROL REGISTER27	PMC27	R/W	H’00	H’0227	8
PORT MODE CONTROL REGISTER28	PMC28	R/W	H’00	H’0228	8
PORT MODE CONTROL REGISTER29	PMC29	R/W	H’00	H’0229	8
PORT MODE CONTROL REGISTER2A	PMC2A	R/W	H’00	H’022A	8
PORT MODE CONTROL REGISTER2B	PMC2B	R/W	H’00	H’022B	8
PORT MODE CONTROL REGISTER2C	PMC2C	R/W	H’00	H’022C	8
PORT MODE CONTROL REGISTER2D	PMC2D	R/W	H’00	H’022D	8
PORT MODE CONTROL REGISTER2E	PMC2E	R/W	H’00	H’022E	8
PORT MODE CONTROL REGISTER2F	PMC2F	R/W	H’00	H’022F	8
PORT MODE CONTROL REGISTER30	PMC30	R/W	H’00	H’0230	8
PORT MODE CONTROL REGISTER31	PMC31	R/W	H’00	H’0231	8
PORT MODE CONTROL REGISTER32	PMC32	R/W	H’00	H’0232	8
PORT MODE CONTROL REGISTER33	PMC33	R/W	H’00	H’0233	8
PORT MODE CONTROL REGISTER34	PMC34	R/W	H’00	H’0234	8
PORT MODE CONTROL REGISTER35	PMC35	R/W	H’00	H’0235	8
PORT MODE CONTROL REGISTER36	PMC36	R/W	H’00	H’0236	8
PORT MODE CONTROL REGISTER37	PMC37	R/W	H’00	H’0237	8
PORT MODE CONTROL REGISTER38	PMC38	R/W	H’00	H’0238	8
PORT MODE CONTROL REGISTER39	PMC39	R/W	H’00	H’0239	8
PORT MODE CONTROL REGISTER3A	PMC3A	R/W	H’00	H’023A	8
PORT MODE CONTROL REGISTER3B	PMC3B	R/W	H’00	H’023B	8

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT MODE CONTROL REGISTER3C	PMC3C	R/W	H'00	H'023C	8
PORT MODE CONTROL REGISTER3D	PMC3D	R/W	H'00	H'023D	8
PORT MODE CONTROL REGISTER3E	PMC3E	R/W	H'00	H'023E	8
PORT MODE CONTROL REGISTER3F	PMC3F	R/W	H'00	H'023F	8
PORT MODE CONTROL REGISTER40	PMC40	R/W	H'00	H'0240	8

41.2.4 Port Function Control Register (PFCm)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT FUNCTION CONTROL REGISTER10	PFC10	R/W	H'0000 0000	H'0440	8/16/32
PORT FUNCTION CONTROL REGISTER11	PFC11	R/W	H'0000 0000	H'0444	8/16/32
PORT FUNCTION CONTROL REGISTER12	PFC12	R/W	H'0000 0000	H'0448	8/16/32
PORT FUNCTION CONTROL REGISTER13	PFC13	R/W	H'0000 0000	H'044C	8/16/32
PORT FUNCTION CONTROL REGISTER14	PFC14	R/W	H'0000 0000	H'0450	8/16/32
PORT FUNCTION CONTROL REGISTER15	PFC15	R/W	H'0000 0000	H'0454	8/16/32
PORT FUNCTION CONTROL REGISTER16	PFC16	R/W	H'0000 0000	H'0458	8/16/32
PORT FUNCTION CONTROL REGISTER17	PFC17	R/W	H'0000 0000	H'045C	8/16/32
PORT FUNCTION CONTROL REGISTER18	PFC18	R/W	H'0000 0000	H'0460	8/16/32
PORT FUNCTION CONTROL REGISTER19	PFC19	R/W	H'0000 0000	H'0464	8/16/32
PORT FUNCTION CONTROL REGISTER1A	PFC1A	R/W	H'0000 0000	H'0468	8/16/32
PORT FUNCTION CONTROL REGISTER1B	PFC1B	R/W	H'0000 0000	H'046C	8/16/32
PORT FUNCTION CONTROL REGISTER1C	PFC1C	R/W	H'0000 0000	H'0470	8/16/32
PORT FUNCTION CONTROL REGISTER1D	PFC1D	R/W	H'0000 0000	H'0474	8/16/32
PORT FUNCTION CONTROL REGISTER1E	PFC1E	R/W	H'0000 0000	H'0478	8/16/32
PORT FUNCTION CONTROL REGISTER1F	PFC1F	R/W	H'0000 0000	H'047C	8/16/32
PORT FUNCTION CONTROL REGISTER20	PFC20	R/W	H'0000 0000	H'0480	8/16/32
PORT FUNCTION CONTROL REGISTER21	PFC21	R/W	H'0000 0000	H'0484	8/16/32
PORT FUNCTION CONTROL REGISTER22	PFC22	R/W	H'0000 0000	H'0488	8/16/32
PORT FUNCTION CONTROL REGISTER23	PFC23	R/W	H'0000 0000	H'048C	8/16/32
PORT FUNCTION CONTROL REGISTER24	PFC24	R/W	H'0000 0000	H'0490	8/16/32
PORT FUNCTION CONTROL REGISTER25	PFC25	R/W	H'0000 0000	H'0494	8/16/32
PORT FUNCTION CONTROL REGISTER26	PFC26	R/W	H'0000 0000	H'0498	8/16/32
PORT FUNCTION CONTROL REGISTER27	PFC27	R/W	H'0000 0000	H'049C	8/16/32
PORT FUNCTION CONTROL REGISTER28	PFC28	R/W	H'0000 0000	H'04A0	8/16/32
PORT FUNCTION CONTROL REGISTER29	PFC29	R/W	H'0000 0000	H'04A4	8/16/32
PORT FUNCTION CONTROL REGISTER2A	PFC2A	R/W	H'0000 0000	H'04A8	8/16/32
PORT FUNCTION CONTROL REGISTER2B	PFC2B	R/W	H'0000 0000	H'04AC	8/16/32
PORT FUNCTION CONTROL REGISTER2C	PFC2C	R/W	H'0000 0000	H'04B0	8/16/32
PORT FUNCTION CONTROL REGISTER2D	PFC2D	R/W	H'0000 0000	H'04B4	8/16/32
PORT FUNCTION CONTROL REGISTER2E	PFC2E	R/W	H'0000 0000	H'04B8	8/16/32
PORT FUNCTION CONTROL REGISTER2F	PFC2F	R/W	H'0000 0000	H'04BC	8/16/32
PORT FUNCTION CONTROL REGISTER30	PFC30	R/W	H'0000 0000	H'04C0	8/16/32
PORT FUNCTION CONTROL REGISTER31	PFC31	R/W	H'0000 0000	H'04C4	8/16/32
PORT FUNCTION CONTROL REGISTER32	PFC32	R/W	H'0000 0000	H'04C8	8/16/32
PORT FUNCTION CONTROL REGISTER33	PFC33	R/W	H'0000 0000	H'04CC	8/16/32
PORT FUNCTION CONTROL REGISTER34	PFC34	R/W	H'0000 0000	H'04D0	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT FUNCTION CONTROL REGISTER35	PFC35	R/W	H’0000 0000	H’04D4	8/16/32
PORT FUNCTION CONTROL REGISTER36	PFC36	R/W	H’0000 0000	H’04D8	8/16/32
PORT FUNCTION CONTROL REGISTER37	PFC37	R/W	H’0000 0000	H’04DC	8/16/32
PORT FUNCTION CONTROL REGISTER38	PFC38	R/W	H’0000 0000	H’04E0	8/16/32
PORT FUNCTION CONTROL REGISTER39	PFC39	R/W	H’0000 0000	H’04E4	8/16/32
PORT FUNCTION CONTROL REGISTER3A	PFC3A	R/W	H’0000 0000	H’04E8	8/16/32
PORT FUNCTION CONTROL REGISTER3B	PFC3B	R/W	H’0000 0000	H’04EC	8/16/32
PORT FUNCTION CONTROL REGISTER3C	PFC3C	R/W	H’0000 0000	H’04F0	8/16/32
PORT FUNCTION CONTROL REGISTER3D	PFC3D	R/W	H’0000 0000	H’04F4	8/16/32
PORT FUNCTION CONTROL REGISTER3E	PFC3E	R/W	H’0000 0000	H’04F8	8/16/32
PORT FUNCTION CONTROL REGISTER3F	PFC3F	R/W	H’0000 0000	H’04FC	8/16/32
PORT FUNCTION CONTROL REGISTER40	PFC40	R/W	H’0000 0000	H’0500	8/16/32

41.2.5 Port Input Register (PINn)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT INPUT REGISTER10	PIN10	R	H’00	H’0810	8
PORT INPUT REGISTER11	PIN11	R	H’00	H’0811	8
PORT INPUT REGISTER12	PIN12	R	H’00	H’0812	8
PORT INPUT REGISTER13	PIN13	R	H’00	H’0813	8
PORT INPUT REGISTER14	PIN14	R	H’00	H’0814	8
PORT INPUT REGISTER15	PIN15	R	H’00	H’0815	8
PORT INPUT REGISTER16	PIN16	R	H’00	H’0816	8
PORT INPUT REGISTER17	PIN17	R	H’00	H’0817	8
PORT INPUT REGISTER18	PIN18	R	H’00	H’0818	8
PORT INPUT REGISTER19	PIN19	R	H’00	H’0819	8
PORT INPUT REGISTER1A	PIN1A	R	H’00	H’081A	8
PORT INPUT REGISTER1B	PIN1B	R	H’00	H’081B	8
PORT INPUT REGISTER1C	PIN1C	R	H’00	H’081C	8
PORT INPUT REGISTER1D	PIN1D	R	H’00	H’081D	8
PORT INPUT REGISTER1E	PIN1E	R	H’00	H’081E	8
PORT INPUT REGISTER1F	PIN1F	R	H’00	H’081F	8
PORT INPUT REGISTER20	PIN20	R	H’00	H’0820	8
PORT INPUT REGISTER21	PIN21	R	H’00	H’0821	8
PORT INPUT REGISTER22	PIN22	R	H’00	H’0822	8
PORT INPUT REGISTER23	PIN23	R	H’00	H’0823	8
PORT INPUT REGISTER24	PIN24	R	H’00	H’0824	8
PORT INPUT REGISTER25	PIN25	R	H’00	H’0825	8
PORT INPUT REGISTER26	PIN26	R	H’00	H’0826	8
PORT INPUT REGISTER27	PIN27	R	H’00	H’0827	8
PORT INPUT REGISTER28	PIN28	R	H’00	H’0828	8
PORT INPUT REGISTER29	PIN29	R	H’00	H’0829	8
PORT INPUT REGISTER2A	PIN2A	R	H’00	H’082A	8
PORT INPUT REGISTER2B	PIN2B	R	H’00	H’082B	8
PORT INPUT REGISTER2C	PIN2C	R	H’00	H’082C	8
PORT INPUT REGISTER2D	PIN2D	R	H’00	H’082D	8

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PORT INPUT REGISTER2E	PIN2E	R	H’00	H’082E	8
PORT INPUT REGISTER2F	PIN2F	R	H’00	H’082F	8
PORT INPUT REGISTER30	PIN30	R	H’00	H’0830	8
PORT INPUT REGISTER31	PIN31	R	H’00	H’0831	8
PORT INPUT REGISTER32	PIN32	R	H’00	H’0832	8
PORT INPUT REGISTER33	PIN33	R	H’00	H’0833	8
PORT INPUT REGISTER34	PIN34	R	H’00	H’0834	8
PORT INPUT REGISTER35	PIN35	R	H’00	H’0835	8
PORT INPUT REGISTER36	PIN36	R	H’00	H’0836	8
PORT INPUT REGISTER37	PIN37	R	H’00	H’0837	8
PORT INPUT REGISTER38	PIN38	R	H’00	H’0838	8
PORT INPUT REGISTER39	PIN39	R	H’00	H’0839	8
PORT INPUT REGISTER3A	PIN3A	R	H’00	H’083A	8
PORT INPUT REGISTER3B	PIN3B	R	H’00	H’083B	8
PORT INPUT REGISTER3C	PIN3C	R	H’00	H’083C	8
PORT INPUT REGISTER3D	PIN3D	R	H’00	H’083D	8
PORT INPUT REGISTER3E	PIN3E	R	H’00	H’083E	8
PORT INPUT REGISTER3F	PIN3F	R	H’00	H’083F	8
PORT INPUT REGISTER40	PIN40	R	H’00	H’0840	8

41.2.6 Interrupt Enable Control Register (ISEL)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
INTERRUPT ENABLE CONTROL REGISTER10	ISEL10	R/W	H’0000 0000	H’2C80	8/16/32
INTERRUPT ENABLE CONTROL REGISTER11	ISEL11	R/W	H’0000 0000	H’2C88	8/16/32
INTERRUPT ENABLE CONTROL REGISTER12	ISEL12	R/W	H’0000 0000	H’2C90	8/16/32
INTERRUPT ENABLE CONTROL REGISTER13	ISEL13	R/W	H’0000 0000	H’2C98	8/16/32
INTERRUPT ENABLE CONTROL REGISTER14	ISEL14	R/W	H’0000 0000	H’2CA0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER15	ISEL15	R/W	H’0000 0000	H’2CA8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER16	ISEL16	R/W	H’0000 0000	H’2CB0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER17	ISEL17	R/W	H’0000 0000	H’2CB8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER18	ISEL18	R/W	H’0000 0000	H’2CC0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER19	ISEL19	R/W	H’0000 0000	H’2CC8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER1A	ISEL1A	R/W	H’0000 0000	H’2CD0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER1B	ISEL1B	R/W	H’0000 0000	H’2CD8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER1C	ISEL1C	R/W	H’0000 0000	H’2CE0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER1D	ISEL1D	R/W	H’0000 0000	H’2CE8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER1E	ISEL1E	R/W	H’0000 0000	H’2CF0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER1F	ISEL1F	R/W	H’0000 0000	H’2CF8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER20	ISEL20	R/W	H’0000 0000	H’2D00	8/16/32
INTERRUPT ENABLE CONTROL REGISTER21	ISEL21	R/W	H’0000 0000	H’2D08	8/16/32
INTERRUPT ENABLE CONTROL REGISTER22	ISEL22	R/W	H’0000 0000	H’2D10	8/16/32
INTERRUPT ENABLE CONTROL REGISTER23	ISEL23	R/W	H’0000 0000	H’2D18	8/16/32
INTERRUPT ENABLE CONTROL REGISTER24	ISEL24	R/W	H’0000 0000	H’2D20	8/16/32
INTERRUPT ENABLE CONTROL REGISTER25	ISEL25	R/W	H’0000 0000	H’2D28	8/16/32
INTERRUPT ENABLE CONTROL REGISTER26	ISEL26	R/W	H’0000 0000	H’2D30	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
INTERRUPT ENABLE CONTROL REGISTER27	ISEL27	R/W	H’0000 0000	H’2D38	8/16/32
INTERRUPT ENABLE CONTROL REGISTER28	ISEL28	R/W	H’0000 0000	H’2D40	8/16/32
INTERRUPT ENABLE CONTROL REGISTER29	ISEL29	R/W	H’0000 0000	H’2D48	8/16/32
INTERRUPT ENABLE CONTROL REGISTER2A	ISEL2A	R/W	H’0000 0000	H’2D50	8/16/32
INTERRUPT ENABLE CONTROL REGISTER2B	ISEL2B	R/W	H’0000 0000	H’2D58	8/16/32
INTERRUPT ENABLE CONTROL REGISTER2C	ISEL2C	R/W	H’0000 0000	H’2D60	8/16/32
INTERRUPT ENABLE CONTROL REGISTER2D	ISEL2D	R/W	H’0000 0000	H’2D68	8/16/32
INTERRUPT ENABLE CONTROL REGISTER2E	ISEL2E	R/W	H’0000 0000	H’2D70	8/16/32
INTERRUPT ENABLE CONTROL REGISTER2F	ISEL2F	R/W	H’0000 0000	H’2D78	8/16/32
INTERRUPT ENABLE CONTROL REGISTER30	ISEL30	R/W	H’0000 0000	H’2D80	8/16/32
INTERRUPT ENABLE CONTROL REGISTER31	ISEL31	R/W	H’0000 0000	H’2D88	8/16/32
INTERRUPT ENABLE CONTROL REGISTER32	ISEL32	R/W	H’0000 0000	H’2D90	8/16/32
INTERRUPT ENABLE CONTROL REGISTER33	ISEL33	R/W	H’0000 0000	H’2D98	8/16/32
INTERRUPT ENABLE CONTROL REGISTER34	ISEL34	R/W	H’0000 0000	H’2DA0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER35	ISEL35	R/W	H’0000 0000	H’2DA8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER36	ISEL36	R/W	H’0000 0000	H’2DB0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER37	ISEL37	R/W	H’0000 0000	H’2DB8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER38	ISEL38	R/W	H’0000 0000	H’2DC0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER39	ISEL39	R/W	H’0000 0000	H’2DC8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER3A_L	ISEL3A_L	R/W	H’0000 0000	H’2DD0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER3A_H	ISEL3A_H	R/W	H’0000 0000	H’2DD4	8/16/32
INTERRUPT ENABLE CONTROL REGISTER3B	ISEL3B	R/W	H’0000 0000	H’2DD8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER3C	ISEL3C	R/W	H’0000 0000	H’2DE0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER3D	ISEL3D	R/W	H’0000 0000	H’2DE8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER3E	ISEL3E	R/W	H’0000 0000	H’2DF0	8/16/32
INTERRUPT ENABLE CONTROL REGISTER3F	ISEL3F	R/W	H’0000 0000	H’2DF8	8/16/32
INTERRUPT ENABLE CONTROL REGISTER40_L	ISEL40_L	R/W	H’0000 0000	H’2E00	8/16/32
INTERRUPT ENABLE CONTROL REGISTER40_H	ISEL40_H	R/W	H’0000 0000	H’2E04	8/16/32

41.2.7 Driving Ability Control Register (IOLH)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DRIVING ABILITY CONTROL REGISTER02	IOLH02	R/W	*1	H’1010	8/16/32
DRIVING ABILITY CONTROL REGISTER03	IOLH03	R/W	*1	H’1018	8/16/32
DRIVING ABILITY CONTROL REGISTER06	IOLH06	R/W	*1	H’1030	8/16/32
DRIVING ABILITY CONTROL REGISTER07_L	IOLH07_L	R/W	*1	H’1038	8/16/32
DRIVING ABILITY CONTROL REGISTER07_H	IOLH07_H	R/W	*1	H’103C	8/16/32
DRIVING ABILITY CONTROL REGISTER08	IOLH08	R/W	*1	H’1040	8/16/32
DRIVING ABILITY CONTROL REGISTER09	IOLH09	R/W	*1	H’1048	8/16/32
DRIVING ABILITY CONTROL REGISTER0A_L	IOLH0A_L	R/W	*1	H’1050	8/16/32
DRIVING ABILITY CONTROL REGISTER0A_H	IOLH0A_H	R/W	*1	H’1054	8/16/32
DRIVING ABILITY CONTROL REGISTER0B_L	IOLH0B_L	R/W	*1	H’1058	8/16/32
DRIVING ABILITY CONTROL REGISTER0B_H	IOLH0B_H	R/W	*1	H’105C	8/16/32
DRIVING ABILITY CONTROL REGISTER0C	IOLH0C	R/W	*1	H’1060	8/16/32
DRIVING ABILITY CONTROL REGISTER0D	IOLH0D	R/W	*1	H’1068	8/16/32
DRIVING ABILITY CONTROL REGISTER10	IOLH10	R/W	*1	H’1080	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DRIVING ABILITY CONTROL REGISTER11	IOLH11	R/W	*1	H'1088	8/16/32
DRIVING ABILITY CONTROL REGISTER12	IOLH12	R/W	*1	H'1090	8/16/32
DRIVING ABILITY CONTROL REGISTER13	IOLH13	R/W	*1	H'1098	8/16/32
DRIVING ABILITY CONTROL REGISTER14	IOLH14	R/W	*1	H'10A0	8/16/32
DRIVING ABILITY CONTROL REGISTER15	IOLH15	R/W	*1	H'10A8	8/16/32
DRIVING ABILITY CONTROL REGISTER16	IOLH16	R/W	*1	H'10B0	8/16/32
DRIVING ABILITY CONTROL REGISTER17	IOLH17	R/W	*1	H'10B8	8/16/32
DRIVING ABILITY CONTROL REGISTER18	IOLH18	R/W	*1	H'10C0	8/16/32
DRIVING ABILITY CONTROL REGISTER19	IOLH19	R/W	*1	H'10C8	8/16/32
DRIVING ABILITY CONTROL REGISTER1A	IOLH1A	R/W	*1	H'10D0	8/16/32
DRIVING ABILITY CONTROL REGISTER1B	IOLH1B	R/W	*1	H'10D8	8/16/32
DRIVING ABILITY CONTROL REGISTER1C	IOLH1C	R/W	*1	H'10E0	8/16/32
DRIVING ABILITY CONTROL REGISTER1D	IOLH1D	R/W	*1	H'10E8	8/16/32
DRIVING ABILITY CONTROL REGISTER1E	IOLH1E	R/W	*1	H'10F0	8/16/32
DRIVING ABILITY CONTROL REGISTER1F	IOLH1F	R/W	*1	H'10F8	8/16/32
DRIVING ABILITY CONTROL REGISTER20	IOLH20	R/W	*1	H'1100	8/16/32
DRIVING ABILITY CONTROL REGISTER21	IOLH21	R/W	*1	H'1108	8/16/32
DRIVING ABILITY CONTROL REGISTER22	IOLH22	R/W	*1	H'1110	8/16/32
DRIVING ABILITY CONTROL REGISTER23	IOLH23	R/W	*1	H'1118	8/16/32
DRIVING ABILITY CONTROL REGISTER36	IOLH36	R/W	*1	H'11B0	8/16/32
DRIVING ABILITY CONTROL REGISTER37	IOLH37	R/W	*1	H'11B8	8/16/32
DRIVING ABILITY CONTROL REGISTER38	IOLH38	R/W	*1	H'11C0	8/16/32
DRIVING ABILITY CONTROL REGISTER39	IOLH39	R/W	*1	H'11C8	8/16/32
DRIVING ABILITY CONTROL REGISTER3A_L	IOLH3A_L	R/W	*1	H'11D0	8/16/32
DRIVING ABILITY CONTROL REGISTER3A_H	IOLH3A_H	R/W	*1	H'11D4	8/16/32
DRIVING ABILITY CONTROL REGISTER3B	IOLH3B	R/W	*1	H'11D8	8/16/32
DRIVING ABILITY CONTROL REGISTER3C	IOLH3C	R/W	*1	H'11E0	8/16/32
DRIVING ABILITY CONTROL REGISTER3D	IOLH3D	R/W	*1	H'11E8	8/16/32
DRIVING ABILITY CONTROL REGISTER3E	IOLH3E	R/W	*1	H'11F0	8/16/32
DRIVING ABILITY CONTROL REGISTER3F	IOLH3F	R/W	*1	H'11F8	8/16/32
DRIVING ABILITY CONTROL REGISTER40_L	IOLH40_L	R/W	*1	H'1200	8/16/32
DRIVING ABILITY CONTROL REGISTER40_H	IOLH40_H	R/W	*1	H'1204	8/16/32

Note 1. For the initial value, refer to **Section 41.3.7, Driving Ability Control Register (IOLH)**.

41.2.8 Slew Rate Switching Register (SR)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
SLEW RATE SWITCHING REGISTER02	SR02	R/W	*1	H'1410	8/16/32
SLEW RATE SWITCHING REGISTER03	SR03	R/W	*1	H'1418	8/16/32
SLEW RATE SWITCHING REGISTER06	SR06	R/W	*1	H'1430	8/16/32
SLEW RATE SWITCHING REGISTER07_L	SR07_L	R/W	*1	H'1438	8/16/32
SLEW RATE SWITCHING REGISTER07_H	SR07_H	R/W	*1	H'143C	8/16/32
SLEW RATE SWITCHING REGISTER08	SR08	R/W	*1	H'1440	8/16/32
SLEW RATE SWITCHING REGISTER09	SR09	R/W	*1	H'1448	8/16/32
SLEW RATE SWITCHING REGISTER0A_L	SR0A_L	R/W	*1	H'1450	8/16/32
SLEW RATE SWITCHING REGISTER0A_H	SR0A_H	R/W	*1	H'1454	8/16/32
SLEW RATE SWITCHING REGISTER0B_L	SR0B_L	R/W	*1	H'1458	8/16/32
SLEW RATE SWITCHING REGISTER0B_H	SR0B_H	R/W	*1	H'145C	8/16/32
SLEW RATE SWITCHING REGISTER0C	SR0C	R/W	*1	H'1460	8/16/32
SLEW RATE SWITCHING REGISTER0D	SR0D	R/W	*1	H'1468	8/16/32
SLEW RATE SWITCHING REGISTER10	SR10	R/W	*1	H'1480	8/16/32
SLEW RATE SWITCHING REGISTER11	SR11	R/W	*1	H'1488	8/16/32
SLEW RATE SWITCHING REGISTER12	SR12	R/W	*1	H'1490	8/16/32
SLEW RATE SWITCHING REGISTER13	SR13	R/W	*1	H'1498	8/16/32
SLEW RATE SWITCHING REGISTER14	SR14	R/W	*1	H'14A0	8/16/32
SLEW RATE SWITCHING REGISTER15	SR15	R/W	*1	H'14A8	8/16/32
SLEW RATE SWITCHING REGISTER16	SR16	R/W	*1	H'14B0	8/16/32
SLEW RATE SWITCHING REGISTER17	SR17	R/W	*1	H'14B8	8/16/32
SLEW RATE SWITCHING REGISTER18	SR18	R/W	*1	H'14C0	8/16/32
SLEW RATE SWITCHING REGISTER19	SR19	R/W	*1	H'14C8	8/16/32
SLEW RATE SWITCHING REGISTER1A	SR1A	R/W	*1	H'14D0	8/16/32
SLEW RATE SWITCHING REGISTER1B	SR1B	R/W	*1	H'14D8	8/16/32
SLEW RATE SWITCHING REGISTER1C	SR1C	R/W	*1	H'14E0	8/16/32
SLEW RATE SWITCHING REGISTER1D	SR1D	R/W	*1	H'14E8	8/16/32
SLEW RATE SWITCHING REGISTER1E	SR1E	R/W	*1	H'14F0	8/16/32
SLEW RATE SWITCHING REGISTER1F	SR1F	R/W	*1	H'14F8	8/16/32
SLEW RATE SWITCHING REGISTER20	SR20	R/W	*1	H'1500	8/16/32
SLEW RATE SWITCHING REGISTER21	SR21	R/W	*1	H'1508	8/16/32
SLEW RATE SWITCHING REGISTER22	SR22	R/W	*1	H'1510	8/16/32
SLEW RATE SWITCHING REGISTER23	SR23	R/W	*1	H'1518	8/16/32
SLEW RATE SWITCHING REGISTER36	SR36	R/W	*1	H'15B0	8/16/32
SLEW RATE SWITCHING REGISTER37	SR37	R/W	*1	H'15B8	8/16/32
SLEW RATE SWITCHING REGISTER38	SR38	R/W	*1	H'15C0	8/16/32
SLEW RATE SWITCHING REGISTER39	SR39	R/W	*1	H'15C8	8/16/32
SLEW RATE SWITCHING REGISTER3A_L	SR3A_L	R/W	*1	H'15D0	8/16/32
SLEW RATE SWITCHING REGISTER3A_H	SR3A_H	R/W	*1	H'15D4	8/16/32
SLEW RATE SWITCHING REGISTER3B	SR3B	R/W	*1	H'15D8	8/16/32
SLEW RATE SWITCHING REGISTER3C	SR3C	R/W	*1	H'15E0	8/16/32
SLEW RATE SWITCHING REGISTER3D	SR3D	R/W	*1	H'15E8	8/16/32
SLEW RATE SWITCHING REGISTER3E	SR3E	R/W	*1	H'15F0	8/16/32
SLEW RATE SWITCHING REGISTER3F	SR3F	R/W	*1	H'15F8	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
SLEW RATE SWITCHING REGISTER40_L	SR40_L	R/W	*1	H'1600	8/16/32
SLEW RATE SWITCHING REGISTER40_H	SR40_H	R/W	*1	H'1604	8/16/32

Note 1. For the initial value, refer to **Section 41.3.8, Slew Rate Switching Register (SR)**.

41.2.9 Pull Up/Pull down Switching Register (PUPD)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PULL UP/PULL DOWN SWITCHING REGISTER10	PUPD10	R/W	H'0000 0000	H'1C80	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER11	PUPD11	R/W	H'0000 0000	H'1C88	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER12	PUPD12	R/W	H'0000 0000	H'1C90	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER13	PUPD13	R/W	H'0000 0000	H'1C98	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER14	PUPD14	R/W	H'0000 0000	H'1CA0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER15	PUPD15	R/W	H'0000 0000	H'1CA8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER16	PUPD16	R/W	H'0000 0000	H'1CB0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER17	PUPD17	R/W	H'0000 0000	H'1CB8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER18	PUPD18	R/W	H'0000 0000	H'1CC0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER19	PUPD19	R/W	H'0000 0000	H'1CC8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER1A	PUPD1A	R/W	H'0000 0000	H'1CD0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER1B	PUPD1B	R/W	H'0000 0000	H'1CD8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER1C	PUPD1C	R/W	H'0000 0000	H'1CE0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER1D	PUPD1D	R/W	H'0000 0000	H'1CE8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER1E	PUPD1E	R/W	H'0000 0000	H'1CF0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER1F	PUPD1F	R/W	H'0000 0000	H'1CF8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER20	PUPD20	R/W	H'0000 0000	H'1D00	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER21	PUPD21	R/W	H'0000 0000	H'1D08	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER22	PUPD22	R/W	H'0000 0000	H'1D10	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER23	PUPD23	R/W	H'0000 0000	H'1D18	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER36	PUPD36	R/W	H'0000 0000	H'1DB0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER37	PUPD37	R/W	H'0000 0000	H'1DB8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER38	PUPD38	R/W	H'0000 0000	H'1DC0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER39	PUPD39	R/W	H'0000 0000	H'1DC8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER3A_L	PUPD3A_L	R/W	H'0000 0000	H'1DD0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER3A_H	PUPD3A_H	R/W	H'0000 0000	H'1DD4	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER3B	PUPD3B	R/W	H'0000 0000	H'1DD8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER3C	PUPD3C	R/W	H'0000 0000	H'1DE0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER3D	PUPD3D	R/W	H'0000 0000	H'1DE8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER3E	PUPD3E	R/W	H'0000 0000	H'1DF0	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER3F	PUPD3F	R/W	H'0000 0000	H'1DF8	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER40_L	PUPD40_L	R/W	H'0000 0000	H'1E00	8/16/32
PULL UP/PULL DOWN SWITCHING REGISTER40_H	PUPD40_H	R/W	H'0000 0000	H'1E04	8/16/32

41.2.10 Write protected register (PWPR)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
WRITE PROTECTED REGISTER	PWPR	R/W	H’80	H’3014	8

41.2.11 Digital Noise Filter Switching Register (FILONOFF)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER SWITCHING REGISTER 01	FILONOFF01	R/W	H’0000 0000	H’2008	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER 10	FILONOFF10	R/W	H’0000 0000	H’2080	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER11	FILONOFF11	R/W	H’0000 0000	H’2088	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER12	FILONOFF12	R/W	H’0000 0000	H’2090	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER13	FILONOFF13	R/W	H’0000 0000	H’2098	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER14	FILONOFF14	R/W	H’0000 0000	H’20A0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER15	FILONOFF15	R/W	H’0000 0000	H’20A8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER16	FILONOFF16	R/W	H’0000 0000	H’20B0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER17	FILONOFF17	R/W	H’0000 0000	H’20B8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER18	FILONOFF18	R/W	H’0000 0000	H’20C0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER19	FILONOFF19	R/W	H’0000 0000	H’20C8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER1A	FILONOFF1A	R/W	H’0000 0000	H’20D0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER1B	FILONOFF1B	R/W	H’0000 0000	H’20D8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER1C	FILONOFF1C	R/W	H’0000 0000	H’20E0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER1D	FILONOFF1D	R/W	H’0000 0000	H’20E8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER1E	FILONOFF1E	R/W	H’0000 0000	H’20F0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER1F	FILONOFF1F	R/W	H’0000 0000	H’20F8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER20	FILONOFF20	R/W	H’0000 0000	H’2100	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER21	FILONOFF21	R/W	H’0000 0000	H’2108	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER22	FILONOFF22	R/W	H’0000 0000	H’2110	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER23	FILONOFF23	R/W	H’0000 0000	H’2118	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER24	FILONOFF24	R/W	H’0000 0000	H’2120	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER25	FILONOFF25	R/W	H’0000 0000	H’2128	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER SWITCHING REGISTER26	FILONOFF26	R/W	H'0000 0000	H'2130	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER27	FILONOFF27	R/W	H'0000 0000	H'2138	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER28	FILONOFF28	R/W	H'0000 0000	H'2140	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER29	FILONOFF29	R/W	H'0000 0000	H'2148	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER2A	FILONOFF2A	R/W	H'0000 0000	H'2150	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER2B	FILONOFF2B	R/W	H'0000 0000	H'2158	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER2C	FILONOFF2C	R/W	H'0000 0000	H'2160	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER2D	FILONOFF2D	R/W	H'0000 0000	H'2168	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER2E	FILONOFF2E	R/W	H'0000 0000	H'2170	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER2F	FILONOFF2F	R/W	H'0000 0000	H'2178	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER30	FILONOFF30	R/W	H'0000 0000	H'2180	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER31	FILONOFF31	R/W	H'0000 0000	H'2188	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER32	FILONOFF32	R/W	H'0000 0000	H'2190	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER33	FILONOFF33	R/W	H'0000 0000	H'2198	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER34	FILONOFF34	R/W	H'0000 0000	H'21A0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER35	FILONOFF35	R/W	H'0000 0000	H'21A8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER36	FILONOFF36	R/W	H'0000 0000	H'21B0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER37	FILONOFF37	R/W	H'0000 0000	H'21B8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER38	FILONOFF38	R/W	H'0000 0000	H'21C0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER39	FILONOFF39	R/W	H'0000 0000	H'21C8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER3A_L	FILONOFF3A_L	R/W	H'0000 0000	H'21D0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER3A_H	FILONOFF3A_H	R/W	H'0000 0000	H'21D4	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER3B	FILONOFF3B	R/W	H'0000 0000	H'21D8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER3C	FILONOFF3C	R/W	H'0000 0000	H'21E0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER3D	FILONOFF3D	R/W	H'0000 0000	H'21E8	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER3E	FILONOFF3E	R/W	H'0000 0000	H'21F0	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER3F	FILONOFF3F	R/W	H'0000 0000	H'21F8	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER SWITCHING REGISTER40_L	FILONOFF40_L	R/W	H'0000 0000	H'2200	8/16/32
DIGITAL NOISE FILTER SWITCHING REGISTER40_H	FILONOFF40_H	R/W	H'0000 0000	H'2204	8/16/32

41.2.12 Digital Noise Filter Number Register (FILNUM)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER NUMBER REGISTER 01	FILNUM01	R/W	H'0000 0000	H'2408	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER 10	FILNUM10	R/W	H'0000 0000	H'2480	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER11	FILNUM11	R/W	H'0000 0000	H'2488	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER12	FILNUM12	R/W	H'0000 0000	H'2490	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER13	FILNUM13	R/W	H'0000 0000	H'2498	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER14	FILNUM14	R/W	H'0000 0000	H'24A0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER15	FILNUM15	R/W	H'0000 0000	H'24A8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER16	FILNUM16	R/W	H'0000 0000	H'24B0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER17	FILNUM17	R/W	H'0000 0000	H'24B8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER18	FILNUM18	R/W	H'0000 0000	H'24C0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER19	FILNUM19	R/W	H'0000 0000	H'24C8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER1A	FILNUM1A	R/W	H'0000 0000	H'24D0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER1B	FILNUM1B	R/W	H'0000 0000	H'24D8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER1C	FILNUM1C	R/W	H'0000 0000	H'24E0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER1D	FILNUM1D	R/W	H'0000 0000	H'24E8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER1E	FILNUM1E	R/W	H'0000 0000	H'24F0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER1F	FILNUM1F	R/W	H'0000 0000	H'24F8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER20	FILNUM20	R/W	H'0000 0000	H'2500	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER21	FILNUM21	R/W	H'0000 0000	H'2508	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER22	FILNUM22	R/W	H'0000 0000	H'2510	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER23	FILNUM23	R/W	H'0000 0000	H'2518	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER24	FILNUM24	R/W	H'0000 0000	H'2520	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER25	FILNUM25	R/W	H'0000 0000	H'2528	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER26	FILNUM26	R/W	H'0000 0000	H'2530	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER27	FILNUM27	R/W	H'0000 0000	H'2538	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER28	FILNUM28	R/W	H'0000 0000	H'2540	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER29	FILNUM29	R/W	H'0000 0000	H'2548	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER2A	FILNUM2A	R/W	H'0000 0000	H'2550	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER2B	FILNUM2B	R/W	H'0000 0000	H'2558	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER2C	FILNUM2C	R/W	H'0000 0000	H'2560	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER2D	FILNUM2D	R/W	H'0000 0000	H'2568	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER2E	FILNUM2E	R/W	H'0000 0000	H'2570	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER2F	FILNUM2F	R/W	H'0000 0000	H'2578	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER30	FILNUM30	R/W	H'0000 0000	H'2580	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER31	FILNUM31	R/W	H'0000 0000	H'2588	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER32	FILNUM32	R/W	H'0000 0000	H'2590	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER33	FILNUM33	R/W	H'0000 0000	H'2598	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER34	FILNUM34	R/W	H'0000 0000	H'25A0	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER NUMBER REGISTER35	FILNUM35	R/W	H’0000 0000	H’25A8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER36	FILNUM36	R/W	H’0000 0000	H’25B0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER37	FILNUM37	R/W	H’0000 0000	H’25B8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER38	FILNUM38	R/W	H’0000 0000	H’25C0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER39	FILNUM39	R/W	H’0000 0000	H’25C8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER3A_L	FILNUM3A_L	R/W	H’0000 0000	H’25D0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER3A_H	FILNUM3A_H	R/W	H’0000 0000	H’25D4	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER3B	FILNUM3B	R/W	H’0000 0000	H’25D8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER3C	FILNUM3C	R/W	H’0000 0000	H’25E0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER3D	FILNUM3D	R/W	H’0000 0000	H’25E8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER3E	FILNUM3E	R/W	H’0000 0000	H’25F0	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER3F	FILNUM3F	R/W	H’0000 0000	H’25F8	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER40_L	FILNUM40_L	R/W	H’0000 0000	H’2600	8/16/32
DIGITAL NOISE FILTER NUMBER REGISTER40_H	FILNUM40_H	R/W	H’0000 0000	H’2604	8/16/32

41.2.13 Digital Noise Filter Clock Selection Register (FILCLKSEL)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 01	FILCLKSEL01	R/W	H’0000 0000	H’2808	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER 10	FILCLKSEL10	R/W	H’0000 0000	H’2880	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER11	FILCLKSEL11	R/W	H’0000 0000	H’2888	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER12	FILCLKSEL12	R/W	H’0000 0000	H’2890	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER13	FILCLKSEL13	R/W	H’0000 0000	H’2898	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER14	FILCLKSEL14	R/W	H’0000 0000	H’28A0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER15	FILCLKSEL15	R/W	H’0000 0000	H’28A8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER16	FILCLKSEL16	R/W	H’0000 0000	H’28B0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER17	FILCLKSEL17	R/W	H’0000 0000	H’28B8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER18	FILCLKSEL18	R/W	H’0000 0000	H’28C0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER19	FILCLKSEL19	R/W	H’0000 0000	H’28C8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER1A	FILCLKSEL1A	R/W	H’0000 0000	H’28D0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER1B	FILCLKSEL1B	R/W	H’0000 0000	H’28D8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER1C	FILCLKSEL1C	R/W	H’0000 0000	H’28E0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER1D	FILCLKSEL1D	R/W	H’0000 0000	H’28E8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER1E	FILCLKSEL1E	R/W	H’0000 0000	H’28F0	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER1F	FILCLKSEL1F	R/W	H'0000 0000	H'28F8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER20	FILCLKSEL20	R/W	H'0000 0000	H'2900	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER21	FILCLKSEL21	R/W	H'0000 0000	H'2908	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER22	FILCLKSEL22	R/W	H'0000 0000	H'2910	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER23	FILCLKSEL23	R/W	H'0000 0000	H'2918	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER24	FILCLKSEL24	R/W	H'0000 0000	H'2920	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER25	FILCLKSEL25	R/W	H'0000 0000	H'2928	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER26	FILCLKSEL26	R/W	H'0000 0000	H'2930	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER27	FILCLKSEL27	R/W	H'0000 0000	H'2938	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER28	FILCLKSEL28	R/W	H'0000 0000	H'2940	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER29	FILCLKSEL29	R/W	H'0000 0000	H'2948	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER2A	FILCLKSEL2A	R/W	H'0000 0000	H'2950	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER2B	FILCLKSEL2B	R/W	H'0000 0000	H'2958	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER2C	FILCLKSEL2C	R/W	H'0000 0000	H'2960	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER2D	FILCLKSEL2D	R/W	H'0000 0000	H'2968	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER2E	FILCLKSEL2E	R/W	H'0000 0000	H'2970	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER2F	FILCLKSEL2F	R/W	H'0000 0000	H'2978	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER30	FILCLKSEL30	R/W	H'0000 0000	H'2980	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER31	FILCLKSEL31	R/W	H'0000 0000	H'2988	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER32	FILCLKSEL32	R/W	H'0000 0000	H'2990	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER33	FILCLKSEL33	R/W	H'0000 0000	H'2998	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER34	FILCLKSEL34	R/W	H'0000 0000	H'29A0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER35	FILCLKSEL35	R/W	H'0000 0000	H'29A8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER36	FILCLKSEL36	R/W	H'0000 0000	H'29B0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER37	FILCLKSEL37	R/W	H'0000 0000	H'29B8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER38	FILCLKSEL38	R/W	H'0000 0000	H'29C0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER39	FILCLKSEL39	R/W	H'0000 0000	H'29C8	8/16/32

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER3A_L	FILCLKSEL3A_L	R/W	H’0000 0000	H’29D0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER3A_H	FILCLKSEL3A_H	R/W	H’0000 0000	H’29D4	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER3B	FILCLKSEL3B	R/W	H’0000 0000	H’29D8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER3C	FILCLKSEL3C	R/W	H’0000 0000	H’29E0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER3D	FILCLKSEL3D	R/W	H’0000 0000	H’29E8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER3E	FILCLKSEL3E	R/W	H’0000 0000	H’29F0	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER3F	FILCLKSEL3F	R/W	H’0000 0000	H’29F8	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER40_L	FILCLKSEL40_L	R/W	H’0000 0000	H’2A00	8/16/32
DIGITAL NOISE FILTER CLOCK SELECTION REGISTER40_H	FILCLKSEL40_H	R/W	H’0000 0000	H’2A04	8/16/32

41.2.14 Ether ch0 IO Voltage Mode Control Register (ETH_ch0)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
ETHER CH0 IO VOLTAGE MODE CONTROL REGISTER	ETH_ch0	R/W	H’00	H’300C	8

41.2.15 Ether ch1 IO Voltage Mode Control Register (ETH_ch1)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
ETHER CH1 IO VOLTAGE MODE CONTROL REGISTER	ETH_ch1	R/W	H’00	H’3010	8

41.2.16 Ether MII/RGMII Mode Control Register (ETH MII/RGMII)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
ETHER MII/RGMII MODE CONTROL REGISTER	ETH MII/RGMII	R/W	H’03	H’3018	8

41.2.17 SD ch0 IO Voltage Mode Control Register (SD_ch0)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
SD CH0 IO VOLTAGE MODE CONTROL REGISTER	SD_ch0	R/W	*1	H’3000	8

Note 1. For the initial value, refer to **Section 41.3.17, SD ch0 IO Voltage Mode Control Register (SD_ch0)**.

41.2.18 SD ch1 IO Voltage Mode Control Register (SD_ch1)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
SD CH1 IO VOLTAGE MODE CONTROL REGISTER	SD_ch1	R/W	H'00	H'3004	8

41.2.19 QSPI IO Voltage Mode Control Register (QSPI)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
QSPI IO VOLTAGE MODE CONTROL REGISTER	QSPI	R/W	*1	H'3008	8

Note 1. For the initial value, refer to **Section 41.3.19, QSPI IO Voltage Mode Control Register (QSPI)**.

41.2.20 Input Enable Control Register (IEN)

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
INPUT ENABLE CONTROL REGISTER02	IEN02	R/W	H'0000 0000	H1810	8/16/32
INPUT ENABLE CONTROL REGISTER04	IEN04	R/W	H'0000 0000	H1820	8/16/32
INPUT ENABLE CONTROL REGISTER06	IEN06	R/W	H'0000 0000	H1830	8/16/32
INPUT ENABLE CONTROL REGISTER07_L	IEN07_L	R/W	H'0000 0000	H1838	8/16/32
INPUT ENABLE CONTROL REGISTER07_H	IEN07_H	R/W	H'0000 0000	H183C	8/16/32
INPUT ENABLE CONTROL REGISTER08	IEN08	R/W	H'0000 0000	H1840	8/16/32
INPUT ENABLE CONTROL REGISTER09	IEN09	R/W	H'0000 0000	H1848	8/16/32
INPUT ENABLE CONTROL REGISTER0E	IEN0E	R/W	H'0000 0000	H1870	8/16/32

41.3 Register Descriptions

41.3.1 Port Register (Pn)

The Pn register sets the GPIO output value.

For the offset address, refer to **Section 41.2.1, Port Register (Pn)**.

Bit	7	6	5	4	3	2	1	0
	P7	P6	P5	P4	P3	P2	P1	P0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	P7	0b	R/W	These bits set the value of the GPIO pin. (P7 to P0) 0b: Low is output. 1b: High is output. Please refer to the table below for the correspondence of each terminal.
6	P6	0b	R/W	
5	P5	0b	R/W	
4	P4	0b	R/W	
3	P3	0b	R/W	
2	P2	0b	R/W	
1	P1	0b	R/W	
0	P0	0b	R/W	

Table 41.1 Correspondence between register and each terminal (1/2)

Bit Name	bit4	bit3	bit2	bit1	bit0
P10	—	—	—	P0_1	P0_0
P11	—	—	—	P1_1	P1_0
P12	—	—	—	P2_1	P2_0
P13	—	—	—	P3_1	P3_0
P14	—	—	—	P4_1	P4_0
P15	—	—	P5_2	P5_1	P5_0
P16	—	—	—	P6_1	P6_0
P17	—	—	P7_2	P7_1	P7_0
P18	—	—	P8_2	P8_1	P8_0
P19	—	—	—	P9_1	P9_0
P1A	—	—	—	P10_1	P10_0
P1B	—	—	—	P11_1	P11_0
P1C	—	—	—	P12_1	P12_0
P1D	—	—	P13_2	P13_1	P13_0
P1E	—	—	—	P14_1	P14_0
P1F	—	—	—	P15_1	P15_0
P20	—	—	—	P16_1	P16_0
P21	—	—	P17_2	P17_1	P17_0
P22	—	—	—	P18_1	P18_0
P23	—	—	—	P19_1	P19_0
P24	—	—	P20_2	P20_1	P20_0
P25	—	—	—	P21_1	P21_0

Table 41.1 Correspondence between register and each terminal (2/2)

Bit Name	bit4	bit3	bit2	bit1	bit0
P26	—	—	—	P22_1	P22_0
P27	—	—	—	P23_1	P23_0
P28	—	—	—	P24_1	P24_0
P29	—	—	—	P25_1	P25_0
P2A	—	—	—	P26_1	P26_0
P2B	—	—	—	P27_1	P27_0
P2C	—	—	—	P28_1	P28_0
P2D	—	—	—	P29_1	P29_0
P2E	—	—	—	P30_1	P30_0
P2F	—	—	—	P31_1	P31_0
P30	—	—	—	P32_1	P32_0
P31	—	—	—	P33_1	P33_0
P32	—	—	—	P34_1	P34_0
P33	—	—	—	P35_1	P35_0
P34	—	—	—	P36_1	P36_0
P35	—	—	P37_2	P37_1	P37_0
P36	—	—	—	P38_1	P38_0
P37	—	—	P39_2	P39_1	P39_0
P38	—	—	P40_2	P40_1	P40_0
P39	—	—	—	P41_1	P41_0
P3A	P42_4	P42_3	P42_2	P42_1	P42_0
P3B	—	P43_3	P43_2	P43_1	P43_0
P3C	—	P44_3	P44_2	P44_1	P44_0
P3D	—	P45_3	P45_2	P45_1	P45_0
P3E	—	P46_3	P46_2	P46_1	P46_0
P3F	—	P47_3	P47_2	P47_1	P47_0
P40	P48_4	P48_3	P48_2	P48_1	P48_0

41.3.2 Port Mode Register (PMn)

The PMn register sets input/output of GPIO.

For the offset address, refer to **Section 41.2.2, Port Mode Register (PMn)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PM4		PM3		PM2		PM1		PM0	
Initial Value	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
9, 8	PM4	00b	R/W	These bits set input/output of the GPIO pin. (PM4 to PM0) 00b: Hi-Z (Initial Value) 01b: Input Mode 10b: Output Mode (Input Disable) 11b: Output Mode (Input Enable) Please refer to the table below for the correspondence of each terminal.
7, 6	PM3	00b	R/W	
5, 4	PM2	00b	R/W	
3, 2	PM1	00b	R/W	
1, 0	PM0	00b	R/W	

Table 41.2 Correspondence between register and each terminal (1/2)

Bit Name	bit9-8	bit7-6	bit5-4	bit3-2	bit1-0
PM10	—	—	—	P0_1	P0_0
PM11	—	—	—	P1_1	P1_0
PM12	—	—	—	P2_1	P2_0
PM13	—	—	—	P3_1	P3_0
PM14	—	—	—	P4_1	P4_0
PM15	—	—	P5_2	P5_1	P5_0
PM16	—	—	—	P6_1	P6_0
PM17	—	—	P7_2	P7_1	P7_0
PM18	—	—	P8_2	P8_1	P8_0
PM19	—	—	—	P9_1	P9_0
PM1A	—	—	—	P10_1	P10_0
PM1B	—	—	—	P11_1	P11_0
PM1C	—	—	—	P12_1	P12_0
PM1D	—	—	P13_2	P13_1	P13_0
PM1E	—	—	—	P14_1	P14_0
PM1F	—	—	—	P15_1	P15_0
PM20	—	—	—	P16_1	P16_0
PM21	—	—	P17_2	P17_1	P17_0
PM22	—	—	—	P18_1	P18_0
PM23	—	—	—	P19_1	P19_0
PM24	—	—	P20_2	P20_1	P20_0
PM25	—	—	—	P21_1	P21_0
PM26	—	—	—	P22_1	P22_0
PM27	—	—	—	P23_1	P23_0
PM28	—	—	—	P24_1	P24_0

Table 41.2 Correspondence between register and each terminal (2/2)

Bit Name	bit9-8	bit7-6	bit5-4	bit3-2	bit1-0
PM29	—	—	—	P25_1	P25_0
PM2A	—	—	—	P26_1	P26_0
PM2B	—	—	—	P27_1	P27_0
PM2C	—	—	—	P28_1	P28_0
PM2D	—	—	—	P29_1	P29_0
PM2E	—	—	—	P30_1	P30_0
PM2F	—	—	—	P31_1	P31_0
PM30	—	—	—	P32_1	P32_0
PM31	—	—	—	P33_1	P33_0
PM32	—	—	—	P34_1	P34_0
PM33	—	—	—	P35_1	P35_0
PM34	—	—	—	P36_1	P36_0
PM35	—	—	P37_2	P37_1	P37_0
PM36	—	—	—	P38_1	P38_0
PM37	—	—	P39_2	P39_1	P39_0
PM38	—	—	P40_2	P40_1	P40_0
PM39	—	—	—	P41_1	P41_0
PM3A	P42_4	P42_3	P42_2	P42_1	P42_0
PM3B	—	P43_3	P43_2	P43_1	P43_0
PM3C	—	P44_3	P44_2	P44_1	P44_0
PM3D	—	P45_3	P45_2	P45_1	P45_0
PM3E	—	P46_3	P46_2	P46_1	P46_0
PM3F	—	P47_3	P47_2	P47_1	P47_0
PM40	P48_4	P48_3	P48_2	P48_1	P48_0

41.3.3 Port Mode Control Register (PMCn)

The PMCn register switches the mode for the multiplexed pins of GPIO.

For the offset address, refer to **Section 41.2.3, Port Mode Control Register (PMCn)**.

Bit	7	6	5	4	3	2	1	0
	PMC7	PMC6	PMC5	PMC4	PMC3	PMC2	PMC1	PMC0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PMC7	0b	R/W	These bits switch the mode for the multiplexed pin of the GPIO pin. (PMC7 to PMC0) 0b: Port Mode (GPIO) 1b: Peripheral Function Mode (Peripheral Function) Please refer to the table below for the correspondence of each terminal.
6	PMC6	0b	R/W	
5	PMC5	0b	R/W	
4	PMC4	0b	R/W	
3	PMC3	0b	R/W	
2	PMC2	0b	R/W	
1	PMC1	0b	R/W	
0	PMC0	0b	R/W	

Table 41.3 Correspondence between register and each terminal (1/2)

Bit Name	bit4	bit3	bit2	bit1	bit0
PMC10	—	—	—	P0_1	P0_0
PMC11	—	—	—	P1_1	P1_0
PMC12	—	—	—	P2_1	P2_0
PMC13	—	—	—	P3_1	P3_0
PMC14	—	—	—	P4_1	P4_0
PMC15	—	—	P5_2	P5_1	P5_0
PMC16	—	—	—	P6_1	P6_0
PMC17	—	—	P7_2	P7_1	P7_0
PMC18	—	—	P8_2	P8_1	P8_0
PMC19	—	—	—	P9_1	P9_0
PMC1A	—	—	—	P10_1	P10_0
PMC1B	—	—	—	P11_1	P11_0
PMC1C	—	—	—	P12_1	P12_0
PMC1D	—	—	P13_2	P13_1	P13_0
PMC1E	—	—	—	P14_1	P14_0
PMC1F	—	—	—	P15_1	P15_0
PMC20	—	—	—	P16_1	P16_0
PMC21	—	—	P17_2	P17_1	P17_0
PMC22	—	—	—	P18_1	P18_0
PMC23	—	—	—	P19_1	P19_0
PMC24	—	—	P20_2	P20_1	P20_0
PMC25	—	—	—	P21_1	P21_0
PMC26	—	—	—	P22_1	P22_0
PMC27	—	—	—	P23_1	P23_0

Table 41.3 Correspondence between register and each terminal (2/2)

Bit Name	bit4	bit3	bit2	bit1	bit0
PMC28	—	—	—	P24_1	P24_0
PMC29	—	—	—	P25_1	P25_0
PMC2A	—	—	—	P26_1	P26_0
PMC2B	—	—	—	P27_1	P27_0
PMC2C	—	—	—	P28_1	P28_0
PMC2D	—	—	—	P29_1	P29_0
PMC2E	—	—	—	P30_1	P30_0
PMC2F	—	—	—	P31_1	P31_0
PMC30	—	—	—	P32_1	P32_0
PMC31	—	—	—	P33_1	P33_0
PMC32	—	—	—	P34_1	P34_0
PMC33	—	—	—	P35_1	P35_0
PMC34	—	—	—	P36_1	P36_0
PMC35	—	—	P37_2	P37_1	P37_0
PMC36	—	—	—	P38_1	P38_0
PMC37	—	—	P39_2	P39_1	P39_0
PMC38	—	—	P40_2	P40_1	P40_0
PMC39	—	—	—	P41_1	P41_0
PMC3A	P42_4	P42_3	P42_2	P42_1	P42_0
PMC3B	—	P43_3	P43_2	P43_1	P43_0
PMC3C	—	P44_3	P44_2	P44_1	P44_0
PMC3D	—	P45_3	P45_2	P45_1	P45_0
PMC3E	—	P46_3	P46_2	P46_1	P46_0
PMC3F	—	P47_3	P47_2	P47_1	P47_0
PMC40	P48_4	P48_3	P48_2	P48_1	P48_0

41.3.4 Port Function Control Register (PFCm)

The PFCm register sets multiplexed functions.

This register can be write-protected by the PWPR register.

For the offset address, refer to **Section 41.2.4, Port Function Control Register (PFCm)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	PFC4		
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PFC3			—	PFC2			—	PFC1			—	PFC0		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19, 15, 11, 7, 3	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
18 to 16	PFC4	000b	R/W	These bits set the multiplexed function of the GPIO pin. (PFC4 to PFC0) 000b: Function0 (Initial value) 001b: Function1 010b: Function2 011b: Function3 100b: Function4 101b: Function5 others: Setting prohibited Please refer to the table below for the correspondence of each terminal.
14 to 12	PFC3	000b	R/W	
10 to 8	PFC2	000b	R/W	
6 to 4	PFC1	000b	R/W	
2 to 0	PFC0	000b	R/W	

Note: Please refer to “Pin function list” for the function of Function (0 to 5).

Table 41.4 Correspondence between register and each terminal (1/2)

Bit Name	bit18-16	bit14-12	bit10-8	bit6-4	bit2-0
PFC10	—	—	—	P0_1	P0_0
PFC11	—	—	—	P1_1	P1_0
PFC12	—	—	—	P2_1	P2_0
PFC13	—	—	—	P3_1	P3_0
PFC14	—	—	—	P4_1	P4_0
PFC15	—	—	P5_2	P5_1	P5_0
PFC16	—	—	—	P6_1	P6_0
PFC17	—	—	P7_2	P7_1	P7_0
PFC18	—	—	P8_2	P8_1	P8_0
PFC19	—	—	—	P9_1	P9_0
PFC1A	—	—	—	P10_1	P10_0
PFC1B	—	—	—	P11_1	P11_0
PFC1C	—	—	—	P12_1	P12_0
PFC1D	—	—	P13_2	P13_1	P13_0
PFC1E	—	—	—	P14_1	P14_0

Table 41.4 Correspondence between register and each terminal (2/2)

Bit Name	bit18-16	bit14-12	bit10-8	bit6-4	bit2-0
PFC1F	—	—	—	P15_1	P15_0
PFC20	—	—	—	P16_1	P16_0
PFC21	—	—	P17_2	P17_1	P17_0
PFC22	—	—	—	P18_1	P18_0
PFC23	—	—	—	P19_1	P19_0
PFC24	—	—	P20_2	P20_1	P20_0
PFC25	—	—	—	P21_1	P21_0
PFC26	—	—	—	P22_1	P22_0
PFC27	—	—	—	P23_1	P23_0
PFC28	—	—	—	P24_1	P24_0
PFC29	—	—	—	P25_1	P25_0
PFC2A	—	—	—	P26_1	P26_0
PFC2B	—	—	—	P27_1	P27_0
PFC2C	—	—	—	P28_1	P28_0
PFC2D	—	—	—	P29_1	P29_0
PFC2E	—	—	—	P30_1	P30_0
PFC2F	—	—	—	P31_1	P31_0
PFC30	—	—	—	P32_1	P32_0
PFC31	—	—	—	P33_1	P33_0
PFC32	—	—	—	P34_1	P34_0
PFC33	—	—	—	P35_1	P35_0
PFC34	—	—	—	P36_1	P36_0
PFC35	—	—	P37_2	P37_1	P37_0
PFC36	—	—	—	P38_1	P38_0
PFC37	—	—	P39_2	P39_1	P39_0
PFC38	—	—	P40_2	P40_1	P40_0
PFC39	—	—	—	P41_1	P41_0
PFC3A	P42_4	P42_3	P42_2	P42_1	P42_0
PFC3B	—	P43_3	P43_2	P43_1	P43_0
PFC3C	—	P44_3	P44_2	P44_1	P44_0
PFC3D	—	P45_3	P45_2	P45_1	P45_0
PFC3E	—	P46_3	P46_2	P46_1	P46_0
PFC3F	—	P47_3	P47_2	P47_1	P47_0
PFC40	P48_4	P48_3	P48_2	P48_1	P48_0

41.3.5 Port Input Register (PINn)

The PINn register is a read-only register to monitor input values of input pins.

For the offset address, refer to **Section 41.2.5, Port Input Register (PINn)**.

Bit	7	6	5	4	3	2	1	0
	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	PIN7	0b	R	Input value of the input pin (PIN7 to PIN0) Please refer to the table below for the correspondence of each terminal.
6	PIN6	0b	R	
5	PIN5	0b	R	
4	PIN4	0b	R	
3	PIN3	0b	R	
2	PIN2	0b	R	
1	PIN1	0b	R	
0	PIN0	0b	R	

Table 41.5 Correspondence between register and each terminal (1/2)

Bit Name	bit4	bit3	bit2	bit1	bit0
PIN10	—	—	—	P0_1	P0_0
PIN11	—	—	—	P1_1	P1_0
PIN12	—	—	—	P2_1	P2_0
PIN13	—	—	—	P3_1	P3_0
PIN14	—	—	—	P4_1	P4_0
PIN15	—	—	P5_2	P5_1	P5_0
PIN16	—	—	—	P6_1	P6_0
PIN17	—	—	P7_2	P7_1	P7_0
PIN18	—	—	P8_2	P8_1	P8_0
PIN19	—	—	—	P9_1	P9_0
PIN1A	—	—	—	P10_1	P10_0
PIN1B	—	—	—	P11_1	P11_0
PIN1C	—	—	—	P12_1	P12_0
PIN1D	—	—	P13_2	P13_1	P13_0
PIN1E	—	—	—	P14_1	P14_0
PIN1F	—	—	—	P15_1	P15_0
PIN20	—	—	—	P16_1	P16_0
PIN21	—	—	P17_2	P17_1	P17_0
PIN22	—	—	—	P18_1	P18_0
PIN23	—	—	—	P19_1	P19_0
PIN24	—	—	P20_2	P20_1	P20_0
PIN25	—	—	—	P21_1	P21_0
PIN26	—	—	—	P22_1	P22_0
PIN27	—	—	—	P23_1	P23_0

Table 41.5 Correspondence between register and each terminal (2/2)

Bit Name	bit4	bit3	bit2	bit1	bit0
PIN28	—	—	—	P24_1	P24_0
PIN29	—	—	—	P25_1	P25_0
PIN2A	—	—	—	P26_1	P26_0
PIN2B	—	—	—	P27_1	P27_0
PIN2C	—	—	—	P28_1	P28_0
PIN2D	—	—	—	P29_1	P29_0
PIN2E	—	—	—	P30_1	P30_0
PIN2F	—	—	—	P31_1	P31_0
PIN30	—	—	—	P32_1	P32_0
PIN31	—	—	—	P33_1	P33_0
PIN32	—	—	—	P34_1	P34_0
PIN33	—	—	—	P35_1	P35_0
PIN34	—	—	—	P36_1	P36_0
PIN35	—	—	P37_2	P37_1	P37_0
PIN36	—	—	—	P38_1	P38_0
PIN37	—	—	P39_2	P39_1	P39_0
PIN38	—	—	P40_2	P40_1	P40_0
PIN39	—	—	—	P41_1	P41_0
PIN3A	P42_4	P42_3	P42_2	P42_1	P42_0
PIN3B	—	P43_3	P43_2	P43_1	P43_0
PIN3C	—	P44_3	P44_2	P44_1	P44_0
PIN3D	—	P45_3	P45_2	P45_1	P45_0
PIN3E	—	P46_3	P46_2	P46_1	P46_0
PIN3F	—	P47_3	P47_2	P47_1	P47_0
PIN40	P48_4	P48_3	P48_2	P48_1	P48_0

41.3.6 Interrupt Enable Control Register (ISEL)

The ISEL register controls whether to enable or disable interrupts.

The pin set as the GPIO input port can be used as an external interrupt input. Controls whether this feature is enabled or disabled. When enabled, this pin can be used as an interrupt by setting the interrupt controller.

For the offset address, refer to **Section 41.2.6, Interrupt Enable Control Register (ISEL)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	ISEL3	—	—	—	—	—	—	—	ISEL2
Initial Value	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ISEL1	—	—	—	—	—	—	—	ISEL0
Initial Value	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25, 23 to 17, 15 to 9, 7 to 1	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
24	ISEL3	0b	R/W	The ISEL register controls whether to enable or disable interrupts. 0b: Disabled (Initial value) 1b: Enabled Please refer to the table below for the correspondence of each terminal.
16	ISEL2	0b	R/W	
8	ISEL1	0b	R/W	
0	ISEL0	0b	R/W	

Note: Before setting this register, disable interrupt detection in the Interrupt Controller.
If PMCn bit = 1, ISEL bit = 1 is prohibited.

Table 41.6 Correspondence between register and each terminal (1/2)

Bit Name	bit24	bit16	bit8	bit0
ISEL10	—	—	P0_1	P0_0
ISEL11	—	—	P1_1	P1_0
ISEL12	—	—	P2_1	P2_0
ISEL13	—	—	P3_1	P3_0
ISEL14	—	—	P4_1	P4_0
ISEL15	—	P5_2	P5_1	P5_0
ISEL16	—	—	P6_1	P6_0
ISEL17	—	P7_2	P7_1	P7_0
ISEL18	—	P8_2	P8_1	P8_0
ISEL19	—	—	P9_1	P9_0
ISEL1A	—	—	P10_1	P10_0
ISEL1B	—	—	P11_1	P11_0
ISEL1C	—	—	P12_1	P12_0
ISEL1D	—	P13_2	P13_1	P13_0
ISEL1E	—	—	P14_1	P14_0

Table 41.6 Correspondence between register and each terminal (2/2)

Bit Name	bit24	bit16	bit8	bit0
ISEL1F	—	—	P15_1	P15_0
ISEL20	—	—	P16_1	P16_0
ISEL21	—	P17_2	P17_1	P17_0
ISEL22	—	—	P18_1	P18_0
ISEL23	—	—	P19_1	P19_0
ISEL24	—	P20_2	P20_1	P20_0
ISEL25	—	—	P21_1	P21_0
ISEL26	—	—	P22_1	P22_0
ISEL27	—	—	P23_1	P23_0
ISEL28	—	—	P24_1	P24_0
ISEL29	—	—	P25_1	P25_0
ISEL2A	—	—	P26_1	P26_0
ISEL2B	—	—	P27_1	P27_0
ISEL2C	—	—	P28_1	P28_0
ISEL2D	—	—	P29_1	P29_0
ISEL2E	—	—	P30_1	P30_0
ISEL2F	—	—	P31_1	P31_0
ISEL30	—	—	P32_1	P32_0
ISEL31	—	—	P33_1	P33_0
ISEL32	—	—	P34_1	P34_0
ISEL33	—	—	P35_1	P35_0
ISEL34	—	—	P36_1	P36_0
ISEL35	—	P37_2	P37_1	P37_0
ISEL36	—	—	P38_1	P38_0
ISEL37	—	P39_2	P39_1	P39_0
ISEL38	—	P40_2	P40_1	P40_0
ISEL39	—	—	P41_1	P41_0
ISEL3A_L	P42_3	P42_2	P42_1	P42_0
ISEL3A_H	—	—	—	P42_4
ISEL3B	P43_3	P43_2	P43_1	P43_0
ISEL3C	P44_3	P44_2	P44_1	P44_0
ISEL3D	P45_3	P45_2	P45_1	P45_0
ISEL3E	P46_3	P46_2	P46_1	P46_0
ISEL3F	P47_3	P47_2	P47_1	P47_0
ISEL40_L	P48_3	P48_2	P48_1	P48_0
ISEL40_H	—	—	—	P48_4

41.3.7 Driving Ability Control Register (IOLH)

This register sets the buffer drive ability of GPIO and Special Purpose Port.

For the offset address, refer to **Section 41.2.7, Driving Ability Control Register (IOLH)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	IOLH3		—	—	—	—	—	—	IOLH2	
Initial Value	—	—	—	—	—	—	*1	*1	—	—	—	—	—	—	*1	*1
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IOLH1		—	—	—	—	—	—	IOLH0	
Initial Value	—	—	—	—	—	—	*1	*1	—	—	—	—	—	—	*1	*1
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
25, 24	IOLH3	*1	R/W	These bits set the drive Ability of each pin. (IOLH4 to IOLH0)
17, 16	IOLH2	*1	R/W	
9, 8	IOLH1	*1	R/W	
1, 0	IOLH0	*1	R/W	

Register Value	Pin Group-A
00b	2 mA
01b (Initial Value)	4 mA
10b	8 mA
11b	12 mA

	Pin Group-B		
	Cload		
Register Value	3.3 V	1.8 V	Output Impedance
00b	5 pF	5 pF	× 0.5 = 100Ω
01b	12 pF	11 pF	× 0.75 = 66Ω
10b (Initial Value)	26 pF	15 pF	× 1.0 = 50Ω
11b	40 pF	21 pF	× 1.5 = 33Ω

Note 1. Refer to the above value for Initial Value.
Please refer to the table below for the correspondence of each pin (including Pin Group).

Table 41.7 Correspondence between register and each terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0	Pin Group
IOLH02	—	—	—	TMS/SWDIO	A
IOLH03	—	—	—	TDO	A
IOLH06	—	SD0_RST#	SD0_CMD	SD0_CLK	B
IOLH07_L	SD0_DATA3	SD0_DATA2	SD0_DATA1	SD0_DATA0	B
IOLH07_H	SD0_DATA7	SD0_DATA6	SD0_DATA5	SD0_DATA4	B
IOLH08	—	—	SD1_CMD	SD1_CLK	B
IOLH09	SD1_DATA3	SD1_DATA2	SD1_DATA1	SD1_DATA0	B
IOLH0A_L	QSPI0_IO2	QSPI0_IO1	QSPI0_IO0	QSPI0_SPCLK	B
IOLH0A_H	—	—	QSPI0_SSL	QSPI0_IO3	B
IOLH0B_L	QSPI1_IO2	QSPI1_IO1	QSPI1_IO0	QSPI1_SPCLK	B
IOLH0B_H	—	—	QSPI1_SSL	QSPI1_IO3	B
IOLH0C	—	—	QSPI_WP#	QSPI_RESET#	B
IOLH0D	—	—	—	WDTOVF_PERROUT#	A
IOLH10	—	—	P0_1	P0_0	A

Table 41.7 Correspondence between register and each terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0	Pin Group
IOLH11	—	—	P1_1	P1_0	A
IOLH12	—	—	P2_1	P2_0	A
IOLH13	—	—	P3_1	P3_0	A
IOLH14	—	—	P4_1	P4_0	A
IOLH15	—	P5_2	P5_1	P5_0	A
IOLH16	—	—	P6_1	P6_0	A
IOLH17	—	P7_2	P7_1	P7_0	A
IOLH18	—	P8_2	P8_1	P8_0	A
IOLH19	—	—	P9_1	P9_0	A
IOLH1A	—	—	P10_1	P10_0	A
IOLH1B	—	—	P11_1	P11_0	A
IOLH1C	—	—	P12_1	P12_0	A
IOLH1D	—	P13_2	P13_1	P13_0	A
IOLH1E	—	—	P14_1	P14_0	A
IOLH1F	—	—	P15_1	P15_0	A
IOLH20	—	—	P16_1	P16_0	A
IOLH21	—	P17_2	P17_1	P17_0	A
IOLH22	—	—	P18_1	P18_0	A
IOLH23	—	—	P19_1	P19_0	A
IOLH36	—	—	P38_1	P38_0	A
IOLH37	—	P39_2	P39_1	P39_0	A
IOLH38	—	P40_2	P40_1	P40_0	A
IOLH39	—	—	P41_1	P41_0	A
IOLH3A_L	P42_3	P42_2	P42_1	P42_0	A
IOLH3A_H	—	—	—	P42_4	A
IOLH3B	P43_3	P43_2	P43_1	P43_0	A
IOLH3C	P44_3	P44_2	P44_1	P44_0	A
IOLH3D	P45_3	P45_2	P45_1	P45_0	A
IOLH3E	P46_3	P46_2	P46_1	P46_0	A
IOLH3F	P47_3	P47_2	P47_1	P47_0	A
IOLH40_L	P48_3	P48_2	P48_1	P48_0	A
IOLH40_H	—	—	—	P48_4	A

41.3.8 Slew Rate Switching Register (SR)

The SR register sets the slew rate of the GPIO pins and Special purpose pins.

For the offset address, refer to **Section 41.2.8, Slew Rate Switching Register (SR)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SR3	—	—	—	—	—	—	—	SR2
Initial Value	—	—	—	—	—	—	—	1	—	—	—	—	—	—	—	1
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SR1	—	—	—	—	—	—	—	SR0
Initial Value	—	—	—	—	—	—	—	1	—	—	—	—	—	—	—	1
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25, 23 to 17, 15 to 9, 7 to 1	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
24	SR3	1b	R/W	These bits control the slew rate of each pin (SR7 to SR0) 0b: Slow 1b: Fast (Initial value) Please refer to the table below for the correspondence of each terminal.
16	SR2	1b	R/W	
8	SR1	1b	R/W	
0	SR0	1b	R/W	

Note: The WDTOVF_PERROUT# bit (H’1468 bit 0) in the SR register is reset by the assertion of the RST_WDTOVFN signal from the CPG.

Table 41.8 Correspondence between register and each terminal (1/2)

Bit Name	bit24	bit16	bit8	bit0
SR02	—	—	—	TMS/SWDIO
SR03	—	—	—	TDO
SR06	—	SD0_RST#	SD0_CMD	SD0_CLK
SR07_L	SD0_DATA3	SD0_DATA2	SD0_DATA1	SD0_DATA0
SR07_H	SD0_DATA7	SD0_DATA6	SD0_DATA5	SD0_DATA4
SR08	—	—	SD1_CMD	SD1_CLK
SR09	SD1_DATA3	SD1_DATA2	SD1_DATA1	SD1_DATA0
SR0A_L	QSPI0_IO2	QSPI0_IO1	QSPI0_IO0	QSPI0_SPCLK
SR0A_H	—	—	QSPI0_SSL	QSPI0_IO3
SR0B_L	QSPI1_IO2	QSPI1_1	QSPI1_IO0	QSPI1_SPCLK
SR0B_H	—	—	QSPI1_SSL	QSPI1_IO3
SR0C	—	QSPI_INT#	QSPI_WP#	QSPI_RESET#
SR0D	—	—	—	WDTOVF_PERROUT#
SR10	—	—	P0_1	P0_0
SR11	—	—	P1_1	P1_0
SR12	—	—	P2_1	P2_0
SR13	—	—	P3_1	P3_0
SR14	—	—	P4_1	P4_0

Table 41.8 Correspondence between register and each terminal (2/2)

Bit Name	bit24	bit16	bit8	bit0
SR15	—	P5_2	P5_1	P5_0
SR16	—	—	P6_1	P6_0
SR17	—	P7_2	P7_1	P7_0
SR18	—	P8_2	P8_1	P8_0
SR19	—	—	P9_1	P9_0
SR1A	—	—	P10_1	P10_0
SR1B	—	—	P11_1	P11_0
SR1C	—	—	P12_1	P12_0
SR1D	—	P13_2	P13_1	P13_0
SR1E	—	—	P14_1	P14_0
SR1F	—	—	P15_1	P15_0
SR20	—	—	P16_1	P16_0
SR21	—	P17_2	P17_1	P17_0
SR22	—	—	P18_1	P18_0
SR23	—	—	P19_1	P19_0
SR36	—	—	P38_1	P38_0
SR37	—	P39_2	P39_1	P39_0
SR38	—	P40_2	P40_1	P40_0
SR39	—	—	P41_1	P41_0
SR3A_L	P42_3	P42_2	P42_1	P42_0
SR3A_H	—	—	—	P42_4
SR3B	P43_3	P43_2	P43_1	P43_0
SR3C	P44_3	P44_2	P44_1	P44_0
SR3D	P45_3	P45_2	P45_1	P45_0
SR3E	P46_3	P46_2	P46_1	P46_0
SR3F	P47_3	P47_2	P47_1	P47_0
SR40_L	P48_3	P48_2	P48_1	P48_0
SR40_H	—	—	—	P48_4

41.3.9 Pull Up/Pull down Switching Register (PUPD)

The PUPD register sets pull-up and pull-down of the GPIO pins and the special purpose pins.

For the offset address, refer to **Section 41.2.9, Pull Up/Pull down Switching Register (PUPD)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PUPD3		—	—	—	—	—	—	PUPD2	
Initial Value	—	—	—	—	—	—	0	0	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PUPD1		—	—	—	—	—	—	PUPD0	
Initial Value	—	—	—	—	—	—	0	0	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
25, 24	PUPD3	00b	R/W	These bits set pull-up and pull-down of each pin (PUPD7 to PUPD0) 00b: Neither pull-up nor pull-down is set. 01b : Pull-up is selected. 10b: Pull-down is selected. 11b: Setting prohibited Please refer to the table below for the correspondence of each terminal.
17, 16	PUPD2	00b	R/W	
9, 8	PUPD1	00b	R/W	
1, 0	PUPD0	00b	R/W	

Table 41.9 Correspondence between register and each terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
PUPD10	—	—	P0_1	P0_0
PUPD11	—	—	P1_1	P1_0
PUPD12	—	—	P2_1	P2_0
PUPD13	—	—	P3_1	P3_0
PUPD14	—	—	P4_1	P4_0
PUPD15	—	P5_2	P5_1	P5_0
PUPD16	—	—	P6_1	P6_0
PUPD17	—	P7_2	P7_1	P7_0
PUPD18	—	P8_2	P8_1	P8_0
PUPD19	—	—	P9_1	P9_0
PUPD1A	—	—	P10_1	P10_0
PUPD1B	—	—	P11_1	P11_0
PUPD1C	—	—	P12_1	P12_0
PUPD1D	—	P13_2	P13_1	P13_0
PUPD1E	—	—	P14_1	P14_0
PUPD1F	—	—	P15_1	P15_0
PUPD20	—	—	P16_1	P16_0
PUPD21	—	P17_2	P17_1	P17_0

Table 41.9 Correspondence between register and each terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
PUPD22	—	—	P18_1	P18_0
PUPD23	—	—	P19_1	P19_0
PUPD36	—	—	P38_1	P38_0
PUPD37	—	P39_2	P39_1	P39_0
PUPD38	—	P40_2	P40_1	P40_0
PUPD39	—	—	P41_1	P41_0
PUPD3A_L	P42_3	P42_2	P42_1	P42_0
PUPD3A_H	—	—	—	P42_4
PUPD3B	P43_3	P43_2	P43_1	P43_0
PUPD3C	P44_3	P44_2	P44_1	P44_0
PUPD3D	P45_3	P45_2	P45_1	P45_0
PUPD3E	P46_3	P46_2	P46_1	P46_0
PUPD3F	P47_3	P47_2	P47_1	P47_0
PUPD40_L	P48_3	P48_2	P48_1	P48_0
PUPD40_H	—	—	—	P48_4

41.3.10 Write protected register (PWPR)

The PWPR register sets write permission / prohibition for the PFC register and the PFCWE bit of this register.

For the offset address, refer to **Section 41.2.10, Write protected register (PWPR)**.

Bit	7	6	5	4	3	2	1	0
	BOWI	PFCWE	—	—	—	—	—	—
Initial Value	1	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	BOWI	1b	R/W	1b: Writing a value to the PFCWE bit disabled 0b: Writing a value to the PFCWE bit enabled
6	PFCWE	0b	R/W	1b: Writing a value to the PFC register enabled 0b: Writing a value to the PFC register disabled
5 to 0	—	All 0	R	Reserved When read, the initial value is read. The written value will be ignored.

41.3.11 Digital Noise Filter Switching Register (FILONOFF)

This register controls whether the digital noise filter is enabled / disabled for GPIO input pin and NMI pin.

For the offset address, refer to **Section 41.2.11, Digital Noise Filter Switching Register (FILONOFF)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FILON3	—	—	—	—	—	—	—	FILON2
Initial Value	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	FILON1	—	—	—	—	—	—	—	FILON0
Initial Value	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25, 23 to 17, 15 to 9, 7 to 1	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
24	FILON3	0b	R/W	These bits set the enable or the disable of Digital noise filter of each pin. (FILON7 to FILON0) 0b: No filter is used. 1b: A filter is used. Please refer to the table below for the correspondence of each terminal.
16	FILON2	0b	R/W	
8	FILON1	0b	R/W	
0	FILON0	0b	R/W	

Note: Before setting this register, disable interrupt detection of the Interrupt controller.

Table 41.10 Correspondence between register and each terminal (1/2)

Bit Name	bit24	bit16	bit8	bit0
FILONOFF01	—	—	—	NMI
FILONOFF10	—	—	P0_1	P0_0
FILONOFF11	—	—	P1_1	P1_0
FILONOFF12	—	—	P2_1	P2_0
FILONOFF13	—	—	P3_1	P3_0
FILONOFF14	—	—	P4_1	P4_0
FILONOFF15	—	P5_2	P5_1	P5_0
FILONOFF16	—	—	P6_1	P6_0
FILONOFF17	—	P7_2	P7_1	P7_0
FILONOFF18	—	P8_2	P8_1	P8_0
FILONOFF19	—	—	P9_1	P9_0
FILONOFF1A	—	—	P10_1	P10_0
FILONOFF1B	—	—	P11_1	P11_0
FILONOFF1C	—	—	P12_1	P12_0
FILONOFF1D	—	P13_2	P13_1	P13_0
FILONOFF1E	—	—	P14_1	P14_0
FILONOFF1F	—	—	P15_1	P15_0
FILONOFF20	—	—	P16_1	P16_0

Table 41.10 Correspondence between register and each terminal (2/2)

Bit Name	bit24	bit16	bit8	bit0
FILONOFF21	—	P17_2	P17_1	P17_0
FILONOFF22	—	—	P18_1	P18_0
FILONOFF23	—	—	P19_1	P19_0
FILONOFF24	—	P20_2	P20_1	P20_0
FILONOFF25	—	—	P21_1	P21_0
FILONOFF26	—	—	P22_1	P22_0
FILONOFF27	—	—	P23_1	P23_0
FILONOFF28	—	—	P24_1	P24_0
FILONOFF29	—	—	P25_1	P25_0
FILONOFF2A	—	—	P26_1	P26_0
FILONOFF2B	—	—	P27_1	P27_0
FILONOFF2C	—	—	P28_1	P28_0
FILONOFF2D	—	—	P29_1	P29_0
FILONOFF2E	—	—	P30_1	P30_0
FILONOFF2F	—	—	P31_1	P31_0
FILONOFF30	—	—	P32_1	P32_0
FILONOFF31	—	—	P33_1	P33_0
FILONOFF32	—	—	P34_1	P34_0
FILONOFF33	—	—	P35_1	P35_0
FILONOFF34	—	—	P36_1	P36_0
FILONOFF35	—	P37_2	P37_1	P37_0
FILONOFF36	—	—	P38_1	P38_0
FILONOFF37	—	P39_2	P39_1	P39_0
FILONOFF38	—	P40_2	P40_1	P40_0
FILONOFF39	—	—	P41_1	P41_0
FILONOFF3A_L	P42_3	P42_2	P42_1	P42_0
FILONOFF3A_H	—	—	—	P42_4
FILONOFF3B	P43_3	P43_2	P43_1	P43_0
FILONOFF3C	P44_3	P44_2	P44_1	P44_0
FILONOFF3D	P45_3	P45_2	P45_1	P45_0
FILONOFF3E	P46_3	P46_2	P46_1	P46_0
FILONOFF3F	P47_3	P47_2	P47_1	P47_0
FILONOFF40_L	P48_3	P48_2	P48_1	P48_0
FILONOFF40_H	—	—	—	P48_4

41.3.12 Digital Noise Filter Number Register (FILNUM)

This register sets the number of FF stages of digital noise filter for GPIO pin and NMI pin.

For the offset address, refer to **Section 41.2.12, Digital Noise Filter Number Register (FILNUM)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	FILNUM3		—	—	—	—	—	—	FILNUM2	
Initial Value	—	—	—	—	—	—	0	0	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FILNUM1		—	—	—	—	—	—	FILNUM0	
Initial Value	—	—	—	—	—	—	0	0	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
25, 24	FILNUM3	00b	R/W	00b: 4-stage filter (41.666 ns x 4 = 166.666 ns) (initial value) 01b: 8-stage filter (41.666 ns x 8 = 333.333 ns) 10b: 12-stage filter (41.666 ns x 12 = 500 ns) 11b: 16-stage filter (41.666 ns x 16 = 666.666 ns) *Note*: This value is the value when the external clock is 24MHz. Please refer to the table below for the correspondence of each terminal.
17, 16	FILNUM2	00b	R/W	
9, 8	FILNUM1	00b	R/W	
1, 0	FILNUM0	00b	R/W	

Note: Before setting this register, disable interrupt detection of the Interrupt controller.

Table 41.11 Correspondence between register and each terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILNUM01	—	—	—	NMI
FILNUM10	—	—	P0_1	P0_0
FILNUM11	—	—	P1_1	P1_0
FILNUM12	—	—	P2_1	P2_0
FILNUM13	—	—	P3_1	P3_0
FILNUM14	—	—	P4_1	P4_0
FILNUM15	—	P5_2	P5_1	P5_0
FILNUM16	—	—	P6_1	P6_0
FILNUM17	—	P7_2	P7_1	P7_0
FILNUM18	—	P8_2	P8_1	P8_0
FILNUM19	—	—	P9_1	P9_0
FILNUM1A	—	—	P10_1	P10_0
FILNUM1B	—	—	P11_1	P11_0
FILNUM1C	—	—	P12_1	P12_0
FILNUM1D	—	P13_2	P13_1	P13_0
FILNUM1E	—	—	P14_1	P14_0

Table 41.11 Correspondence between register and each terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILNUM1F	—	—	P15_1	P15_0
FILNUM20	—	—	P16_1	P16_0
FILNUM21	—	P17_2	P17_1	P17_0
FILNUM22	—	—	P18_1	P18_0
FILNUM23	—	—	P19_1	P19_0
FILNUM24	—	P20_2	P20_1	P20_0
FILNUM25	—	—	P21_1	P21_0
FILNUM26	—	—	P22_1	P22_0
FILNUM27	—	—	P23_1	P23_0
FILNUM28	—	—	P24_1	P24_0
FILNUM29	—	—	P25_1	P25_0
FILNUM2A	—	—	P26_1	P26_0
FILNUM2B	—	—	P27_1	P27_0
FILNUM2C	—	—	P28_1	P28_0
FILNUM2D	—	—	P29_1	P29_0
FILNUM2E	—	—	P30_1	P30_0
FILNUM2F	—	—	P31_1	P31_0
FILNUM30	—	—	P32_1	P32_0
FILNUM31	—	—	P33_1	P33_0
FILNUM32	—	—	P34_1	P34_0
FILNUM33	—	—	P35_1	P35_0
FILNUM34	—	—	P36_1	P36_0
FILNUM35	—	P37_2	P37_1	P37_0
FILNUM36	—	—	P38_1	P38_0
FILNUM37	—	P39_2	P39_1	P39_0
FILNUM38	—	P40_2	P40_1	P40_0
FILNUM39	—	—	P41_1	P41_0
FILNUM3A_L	P42_3	P42_2	P42_1	P42_0
FILNUM3A_H	—	—	—	P42_4
FILNUM3B	P43_3	P43_2	P43_1	P43_0
FILNUM3C	P44_3	P44_2	P44_1	P44_0
FILNUM3D	P45_3	P45_2	P45_1	P45_0
FILNUM3E	P46_3	P46_2	P46_1	P46_0
FILNUM3F	P47_3	P47_2	P47_1	P47_0
FILNUM40_L	P48_3	P48_2	P48_1	P48_0
FILNUM40_H	—	—	—	P48_4

41.3.13 Digital Noise Filter Clock Selection Register (FILCLKSEL)

The FILCLKSEL register selects the divided clock to be input to digital noise filters.

For the offset address, refer to **Section 41.2.13, Digital Noise Filter Clock Selection Register (FILCLKSEL)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	FILCLK3		—	—	—	—	—	—	FILCLK2	
Initial Value	—	—	—	—	—	—	0	0	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FILCLK1		—	—	—	—	—	—	FILCLK0	
Initial Value	—	—	—	—	—	—	0	0	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26, 23 to 18, 15 to 10, 7 to 2	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
25, 24	FILCLK3	00b	R/W	00b: Not divided (initial value) 01b: Divided by 9000 (41.666 ns x 9000 = 375,000 ns) 10b: Divided by 18000 (41.666 ns x 18000 = 750,000 ns) 11b: Divided by 36000 (41.666 ns x 36000 = 1,500,000 ns) *Note*: This value is the value when the external clock is 24MHz. Please refer to the table below for the correspondence of each terminal.
17, 16	FILCLK2	00b	R/W	
9, 8	FILCLK1	00b	R/W	
1, 0	FILCLK0	00b	R/W	

Note: Before setting this register, disable interrupt detection of the Interrupt controller.

Table 41.12 Correspondence between register and each terminal (1/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILCLKSEL01	—	—	—	NMI
FILCLKSEL10	—	—	P0_1	P0_0
FILCLKSEL11	—	—	P1_1	P1_0
FILCLKSEL12	—	—	P2_1	P2_0
FILCLKSEL13	—	—	P3_1	P3_0
FILCLKSEL14	—	—	P4_1	P4_0
FILCLKSEL15	—	P5_2	P5_1	P5_0
FILCLKSEL16	—	—	P6_1	P6_0
FILCLKSEL17	—	P7_2	P7_1	P7_0
FILCLKSEL18	—	P8_2	P8_1	P8_0
FILCLKSEL19	—	—	P9_1	P9_0
FILCLKSEL1A	—	—	P10_1	P10_0
FILCLKSEL1B	—	—	P11_1	P11_0
FILCLKSEL1C	—	—	P12_1	P12_0
FILCLKSEL1D	—	P13_2	P13_1	P13_0
FILCLKSEL1E	—	—	P14_1	P14_0

Table 41.12 Correspondence between register and each terminal (2/2)

Bit Name	bit25-24	bit17-16	bit9-8	bit1-0
FILCLKSEL1F	—	—	P15_1	P15_0
FILCLKSEL20	—	—	P16_1	P16_0
FILCLKSEL21	—	P17_2	P17_1	P17_0
FILCLKSEL22	—	—	P18_1	P18_0
FILCLKSEL23	—	—	P19_1	P19_0
FILCLKSEL24	—	P20_2	P20_1	P20_0
FILCLKSEL25	—	—	P21_1	P21_0
FILCLKSEL26	—	—	P22_1	P22_0
FILCLKSEL27	—	—	P23_1	P23_0
FILCLKSEL28	—	—	P24_1	P24_0
FILCLKSEL29	—	—	P25_1	P25_0
FILCLKSEL2A	—	—	P26_1	P26_0
FILCLKSEL2B	—	—	P27_1	P27_0
FILCLKSEL2C	—	—	P28_1	P28_0
FILCLKSEL2D	—	—	P29_1	P29_0
FILCLKSEL2E	—	—	P30_1	P30_0
FILCLKSEL2F	—	—	P31_1	P31_0
FILCLKSEL30	—	—	P32_1	P32_0
FILCLKSEL31	—	—	P33_1	P33_0
FILCLKSEL32	—	—	P34_1	P34_0
FILCLKSEL33	—	—	P35_1	P35_0
FILCLKSEL34	—	—	P36_1	P36_0
FILCLKSEL35	—	P37_2	P37_1	P37_0
FILCLKSEL36	—	—	P38_1	P38_0
FILCLKSEL37	—	P39_2	P39_1	P39_0
FILCLKSEL38	—	P40_2	P40_1	P40_0
FILCLKSEL39	—	—	P41_1	P41_0
FILCLKSEL3A_L	P42_3	P42_2	P42_1	P42_0
FILCLKSEL3A_H	—	—	—	P42_4
FILCLKSEL3B	P43_3	P43_2	P43_1	P43_0
FILCLKSEL3C	P44_3	P44_2	P44_1	P44_0
FILCLKSEL3D	P45_3	P45_2	P45_1	P45_0
FILCLKSEL3E	P46_3	P46_2	P46_1	P46_0
FILCLKSEL3F	P47_3	P47_2	P47_1	P47_0
FILCLKSEL40_L	P48_3	P48_2	P48_1	P48_0
FILCLKSEL40_H	—	—	—	P48_4

The operation of the digital noise filter will be explained using a timing chart using the external terminal P0_0 as an example.

(1) When the digital noise filter is disabled (FILONOFF = 0)

The timing chart of FILONOFF = 0 (OFF), FILNUM [1: 0] = “00” (FF4 stage), FILCLKSEL [1: 0] = “00” (without frequency division) is shown below.

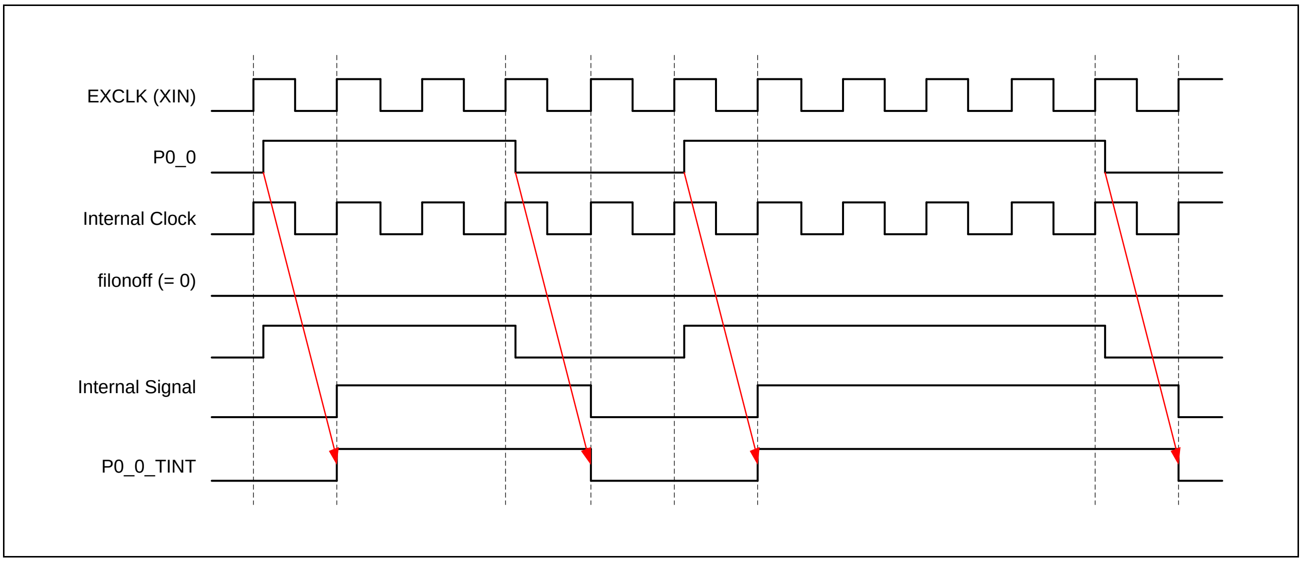


Figure 41.14 Digital noise filter (FILONOFF = 0) timing chart

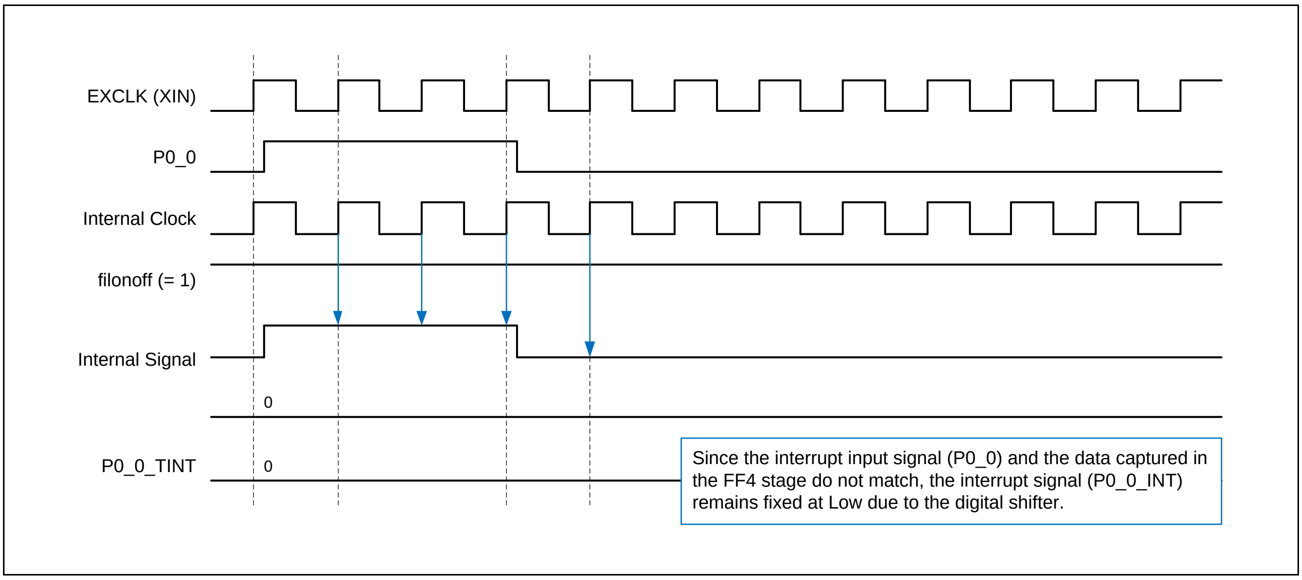
The signal at the external terminal (P0_0) is not filtered regardless of the number of stages set in the FILNUM register.

The digital noise filter circuit inserts one stage of FF in the output of a digital noise filter.

Therefore, the external terminal (P0_0) signal is propagated internally with a delay of one cycle. (EXCLK (XIN)).

(2) When the digital noise filter is enabled (FILONOFF = 1)

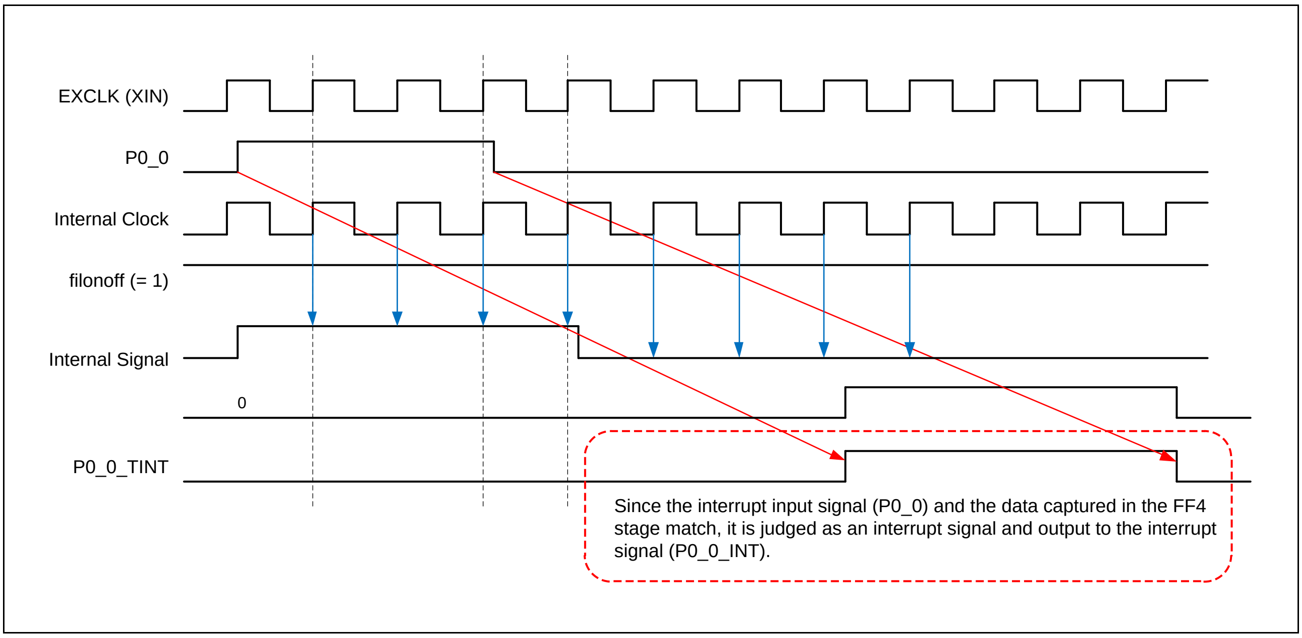
The timing chart of FILONOFF = 1 (ON), FILNUM [1: 0] = “00” (FF4 stage), FILCLKSEL [1: 0] = “00” (without frequency division) is shown below.



Since the interrupt input signal (P0_0) and the data captured in the FF4 stage do not match, the interrupt signal (P0_0_INT) remains fixed at Low due to the digital shifter.

Figure 41.15 Digital noise filter (FILONOFF = 1) Timing chart (1)

The signal of the external terminal (P0_0) is filtered, if even one signal captured by each FF for the number of FF stages set in the FILNUM register does not match.



Since the interrupt input signal (P0_0) and the data captured in the FF4 stage match, it is judged as an interrupt signal and output to the interrupt signal (P0_0_INT).

Figure 41.16 Digital noise filter (FILONOFF = 1) Timing chart (2)

The external terminal (P0_0) signal is the signal captured by each FF for the number of FF stages set in the FILNUM register. If they all match, no filtering is done.

The digital noise filter circuit outputs to the interrupt signal with a delay of FILTER_CYCLE.

[FILTER_CYCLE Calculation]

FILTER_CYCLE = FILNUM_FF + $FILTER_FF

FILNUM_FF: Number of FF stages set in the FILNUM register @EXCLK(XIN)
FILTER_FF: FF 2 stage in digital noise filter @EXCLK(XIN) + Digital noise filter output FF 1 stage @EXCLK(XIN)

Note: One stage of FF is inserted in the final stage of the digital noise filter circuit. Therefore, interrupt signals less than 1CLK @ EXCLK (XIN) may be removed regardless of whether the digital noise filter is enabled or disabled.

41.3.14 Ether ch0 Voltage Mode Control Register (ETH_ch0)

This register is sets the IO voltage mode control of Ether ch0.

For the offset address, refer to **Section 41.2.14, Ether ch0 IO Voltage Mode Control Register (ETH_ch0)**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ETH0_2.5V_PVDD	ETH0_1.8V_PVDD
Initial Value	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>7 to 2</td><td>—</td><td>—</td><td>R</td><td>Reserved
When read, the initial value is read. The written value will be ignored.</td></tr>
<tr><td>1</td><td>ETH0_2.5V_PVDD</td><td>0b</td><td>R/W</td><td rowspan="2">These bits set the IO voltage mode.<table><tr><th>Bit 1</th><th>Bit 0</th><th>IO Voltage</th></tr><tr><td>0</td><td>0</td><td>3.3V (Initial value)</td></tr><tr><td>0</td><td>1</td><td>1.8V</td></tr><tr><td>1</td><td>0</td><td>2.5V</td></tr><tr><td>1</td><td>1</td><td>Setting prohibited</td></tr></table></td></tr>
<tr><td>0</td><td>ETH0_1.8V_PVDD</td><td>0b</td><td>R/W</td></tr>
</table>

- For the IO voltage mode of P20_0 to P28_1, set the above register according to the voltage mode of the external device to be connected.
- At Function1, the Ether ch0 function is selected, so set the value of this register according to the power supply voltage (PVDD182533_0) to be used.

  When Function2 to Function5, a function other than Ether ch0 is selected, so set the value of this register to IO voltage mode (3.3v).

  For each peripheral function, refer to Function0 to Function5 in the Pin function List.

  For Function2 to Function5, only 3.3V can be set.

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3V and the register is set to 1.8V, the reliability of this LSI will be affected.

41.3.15 Ether ch1 Voltage Mode Control Register (ETH_ch1)

This register sets the IO voltage mode control for Etherch1.

For the offset address, refer to **Section 41.2.15, Ether ch1 IO Voltage Mode Control Register (ETH_ch1)**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ETH1_2.5V_PVDD	ETH1_1.8V_PVDD
Initial Value	—	—	—	—	—	—	0	0
R/W	R	R	R	R	R	R	R/W	R/W

<table>
<tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr>
<tr><td>7 to 2</td><td>—</td><td>—</td><td>R</td><td>Reserved
When read, the initial value is read. The written value will be ignored.</td></tr>
<tr><td>1</td><td>ETH1_2.5V_PVDD</td><td>0b</td><td>R/W</td><td rowspan="2">These bits set the IO voltage mode.
<table>
<tr><th>Bit 1</th><th>Bit 0</th><th>IO Voltage</th></tr>
<tr><td>0</td><td>0</td><td>3.3V (Initial value)</td></tr>
<tr><td>0</td><td>1</td><td>1.8V</td></tr>
<tr><td>1</td><td>0</td><td>2.5V</td></tr>
<tr><td>1</td><td>1</td><td>Setting prohibited</td></tr>
</table></td></tr>
<tr><td>0</td><td>ETH1_1.8V_PVDD</td><td>0b</td><td>R/W</td></tr>
</table>

- For the IO voltage mode of P29_0 to P37_2, set the above register according to the voltage mode of the external device to be connected.
- At Function1, the Ether ch1 function is selected, so set the value of this register according to the power supply voltage (PVDD182533_1) to be used.

  When Function2 to Function5, a function other than Ether ch1 is selected, so set the value of this register to IO voltage mode (3.3v).

  For each peripheral function, refer to Function0 to Function5 in the Pin function List.

  For Function2 to Function5, only 3.3V can be set.

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3V and the register is set to 1.8V, the reliability of this LSI will be affected.

41.3.16 Ether MII/RGMII Mode Control Register (ETH MII/RGMII)

This register can be selected “MII” or “RGMII” of Ether.

This register is valid only for Function1

For the offset address, refer to **Section 41.2.16, Ether MII/RGMII Mode Control Register (ETH MII/RGMII)**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ETH1_ mode	ETH0_ mode
Initial Value	—	—	—	—	—	—	1	1
R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
1	ETH1_mode	1b	R/W	Ether MII or RGMII can be selected by setting this register. 0b = RGMII The Direction of the IO buffer is Output. 1b = MII(Initial value) The Direction of the IO buffer is Input.
0	ETH0_mode	1b	R/W	

Note: By setting this register, the direction of some pins when using Ether’s MII or RGMII can be selected. Refer to **Section 30, Gigabit Ethernet Interface** chapter for switching between MII and RGMII of Ether.

Note: In RGMII mode, set the external terminals shown below to port mode (PMCn = 0) and I/O disabled (PMn = 00b).

Ether ch0		Ether ch1	
Terminal name Function0	Terminal name Function1	Terminal name Function0	Terminal name Function1
P23_0	ET0_TX_COL	P32_0	ET1_TX_COL
P23_1	ET0_TX_CRS	P33_1	ET1_TX_CRS
P27_0	ET0_RX_ERR	P36_0	ET1_RX_ERR

41.3.17 SD ch0 IO Voltage Mode Control Register (SD_ch0)

This register sets the IO voltage mode control of SD ch0.

For the offset address, refer to **Section 41.2.17, SD ch0 IO Voltage Mode Control Register (SD_ch0)**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SD0_PVDD
Initial Value	—	—	—	—	—	—	—	0[*1]
R/W	R	R	R	R	R	R	R	R/W[*1]

Note 1. In Boot Mode1 and Boot Mode2, R/W is R (Read Only).

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
0	SD0_PVDD	0b[*1]	R/W	This bit sets the IO voltage mode. 1b: I/O domain voltage ≤ 1.8V 0b: I/O domain voltage ≥ 3.3V (Initial value)[*1]

Note 1. The initial value of this register

- The IO voltage mode of SD0_CLK, SD0_CMD, SD0_DATA [7-0], SD0_RST# is set by selecting each mode of this register and MD_BOOT [2: 0].
  - When Boot Modes 1 and 2 are not selected: This register can be Read/Write (R/W). The initial value is the above value.
  - When Boot Modes 1 and 2 are selected: This register is Read Only (R), and the voltage mode of the selected device is as follows.

MD_BOOT[2:0]			Mode	Selection function	Selection Device	Control method
0	0	1	Boot Mode 1	SDHI0	eMMC(1.8V)	By Hardware
0	1	0	Boot Mode 2	SDHI0	eMMC(3.3V)	By Hardware
Other			SD_ch0 Control by Register			

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3V and the register is set to 1.8V, the reliability of this LSI will be affected.

41.3.18 SD ch1 IO Voltage Mode Control Register (SD_ch1)

This register sets the IO voltage mode control of SD ch1.

For the offset address, refer to **Section 41.2.18, SD ch1 IO Voltage Mode Control Register (SD_ch1)**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SD1_PVDD
Initial Value	—	—	—	—	—	—	—	0
R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
0	SD1_PVDD	0b	R/W	This bit sets the IO voltage mode. 1b: I/O domain voltage ≤ 1.8V 0b: I/O domain voltage ≥ 3.3V (Initial value)

Note: The IO voltage mode of SD1_CLK, SD1_CMD, SD1_DATA [3-0] is set by this register.

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 3.3V and the register is set to 1.8V, the reliability of this LSI will be affected.

41.3.19 QSPI IO Voltage Mode Control Register (QSPI)

This register sets the IO voltage mode control of QSPI.

For the offset address, refer to **Section 41.2.19, QSPI IO Voltage Mode Control Register (QSPI)**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	QSPI_PVDD
Initial Value	—	—	—	—	—	—	—	0[*1]
R/W	R	R	R	R	R	R	R	R/W[*1]

Note 1. In Boot Mode3 and Boot Mode4, R/W is R (Read Only).

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
0	QSPI_PVDD	0b[*1]	R/W	This bit sets the IO voltage mode. 1b: I/O domain voltage ≤ 1.8V 0b: I/O domain voltage ≥ 3.3V (Initial value)[*1]

Note 1. The initial value of this register

- The IO voltage of QSPI0_SPCLK, QSPI0_IO[3-0], QSPI0_SSL, QSPI1_SPCLK, QSPI1_IO [3-0], QSPI1_SSL, QSPI_RESET#, QSPI_WP#, QSPI_INT# is set by each mode of this register and MD_BOOT[2: 0].
  - When Boot Modes 3 and 4 are not selected: This register can be Read/Write (R/W). The initial value is the above value.
  - When Boot Modes 3 and 4 are selected: This register is Read Only (R), and the voltage mode of the selected device is as follows.

MD_BOOT[2:0]			Mode	Selection Function	Selection Device	Control method
0	1	1	Boot Mode 3	SPI Multi I/O Bus Controller	Single/Quad/Octal(1.8V)	By Hardware
1	0	0	Boot Mode 4		Single/Quad(3.3V)	By Hardware
Other			QSPI Control by register			

Note: When setting this register, do not set the IO voltage mode lower than the applied voltage. For example, if the applied voltage is set to 1.8V and the register is set to 3.3V, the reliability of this LSI will be affected.

41.3.20 Input Enable Control Register (IEN)

This register is a register that controls the input of the special purpose port.

For the offset address, refer to **Section 41.2.20, Input Enable Control Register (IEN)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	IEN3	—	—	—	—	—	—	—	IEN2
Initial Value	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IEN1	—	—	—	—	—	—	—	IEN0
Initial Value	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25, 23 to 17, 15 to 9, 7 to 1	—	—	R	Reserved When read, the initial value is read. The written value will be ignored.
24	IEN3	0b	R/W	These bits control inputs of p bit of a pin with a multiplexed function (except for GPIO) and a dedicated pin (p = 0 to 7) 0b: Input disabled (Initial value) 1b: Input enabled Please refer to the table below for the correspondence of each terminal.
16	IEN2	0b	R/W	
8	IEN1	0b	R/W	
0	IEN0	0b	R/W	

Table 41.13 Correspondence between register and each terminal

Bit Name	bit24	bit16	bit8	bit0
IEN02	—	—	—	TMS/SWDIO
IEN04	—	—	AUDIO_CLK2	AUDIO_CLK1
IEN06	—	—	SD0_CMD	—
IEN07_L	SD0_DATA3	SD0_DATA2	SD0_DATA1	SD0_DATA0
IEN07_H	SD0_DATA7	SD0_DATA6	SD0_DATA5	SD0_DATA4
IEN08	—	—	SD1_CMD	—
IEN09	SD1_DATA3	SD1_DATA2	SD1_DATA1	SD1_DATA0
IEN0E	RIIC1_SCL	RIIC1_SDA	RIIC0_SCL	RIIC0_SDA

41.4 Operation

The operating procedure for the general-purpose I/O port function and the operating procedure for peripheral functions are shown below.

41.4.1 Operation for GPIO Function

(1) Set the PMCn register to “0” (initial value = 0).

(2) Set the GPIO input / output with the PMn register.

(3) Set the GPIO output value in the Pn register.

(4) The input value of the input terminal can be detected by the PINn register.

(5) When setting the interrupt input using the GPIO input pin, set the ISEL register to “1”.

- Set the FILONOFF, IOLH, SR, and PUPD registers between (1) and (2) as necessary.
- Set the FILONOFF register when using GPIO to set a digital noise filter for interrupt input and GPIO input.
- Set the FILNUM and FILCLKSEL registers according to the purpose of use.

41.4.2 Operation for Peripheral Function

(1) Set the PMCn register to “1” (initial value = 0).

(2) Set the PWPR register to allow writing to the PFCm register.

After setting the PWPR.B0WI bit to “0” (initial value = 1), set the PWPR.PFCWE bit to “1” (initial value = 0).

Select the required function from Functions 1-5. (Hereafter, Function1 setting example)

(3) Set PFCm = 001b and switch to Function1.

(4) Set the PFCm register to write-protected. After setting the PWPR.PFCWE bit to “0”, set the PWPR.B0WI bit to “1”.

- Set the FILONOFF, IOLH, SR, and PUPD registers between (1) and (2) as necessary.
- Set the FILONOFF register when using GPIO to set a digital noise filter for the IRQ[0-7].
- Set the FILNUM and FILCLKSEL registers according to the purpose of use. And also, the NMI input can set the noise filter by the same registers.

42. Low Power Mode

This chapter describes the lower mode of this LSI.

42.1 Overview

This LSI has three modes of low power consumption shown in the following table.

These three modes work independently.

Table 42.1 Low Power Mode

Low Power Mode	Description	Return factor
Module Standby Mode	Stop the clock of specific peripheral module by setting the CPG register. Refer to **Section 42.2, Module Standby Mode** for detail.	CPG register setting
Cortex-M33 Sleep Mode	Put Cortex-M33 into a low power state by issuing a WFI instruction. The SLEEPDEEP bit of SCR register should be set to 1. Refer to **Section 3.4.4, Cortex-M33 Sleep Mode** and **Section 42.3, Cortex-M33 Sleep Mode** for detail.	Interrupt SysTick Timer
Cortex-A55 Sleep Mode	Put Cortex-A55 into a low power state by issuing a WFI instruction. The WFI instruction can be issued each core. Refer to **Section 2.3.1, Cortex-A55 Sleep Mode** and **Section 42.4, Cortex-A55 Sleep Mode** for detail.	Interrupt

Note: When a system reset is applied, the LSI is released from the low power mode and initialized.

42.2 Module Standby Mode

42.2.1 Overview

The Module Standby Mode is a mode that requests the clock stop of the module specified by the master.

The purpose of this mode is to reduce power consumption by stopping unnecessary functions.

The master which can request the clock stop is Cortex-A55*[1] and Cortex-M33. So, Cortex-A55 and Cortex-M33 cannot be placed in the Module Standby Mode. To reduce the power consumption in the CPUs, use the CPU sleep modes to be described later.

Note 1. In case of Cortex-A55 dual core product, the master is Cortex-A55 core0 and Cortex-A55 core1.

This mode is implemented by stopping the clock supply by the CPG register and switching the MSTOP signal of the bus through CPG register settings.

If the master accesses a module that has the clock stopped and the MSTOP bit set, a bus error will occur.

The following registers are used in the Module Standby Mode.

Table 42.2 Module Standby Mode related registers

Register name	Abbreviation	Description
Clock Control Register	CPG_CLKON_***	The detail information of each register is described in **Section 6, System Controller (SYSC)** and **Section 7, Clock Pulse Generator (CPG)**. The usage of these registers is described in **Section 42.2.2, Operation**.
MSTOP Register	CPG_BUS_***_MSTOP CPG_MHU_STOP	
GPU Lowpower Sequence Control Register	SYS_LP_GPU_CTL	

The following figure shows the block connection overview related Module Standby Mode.

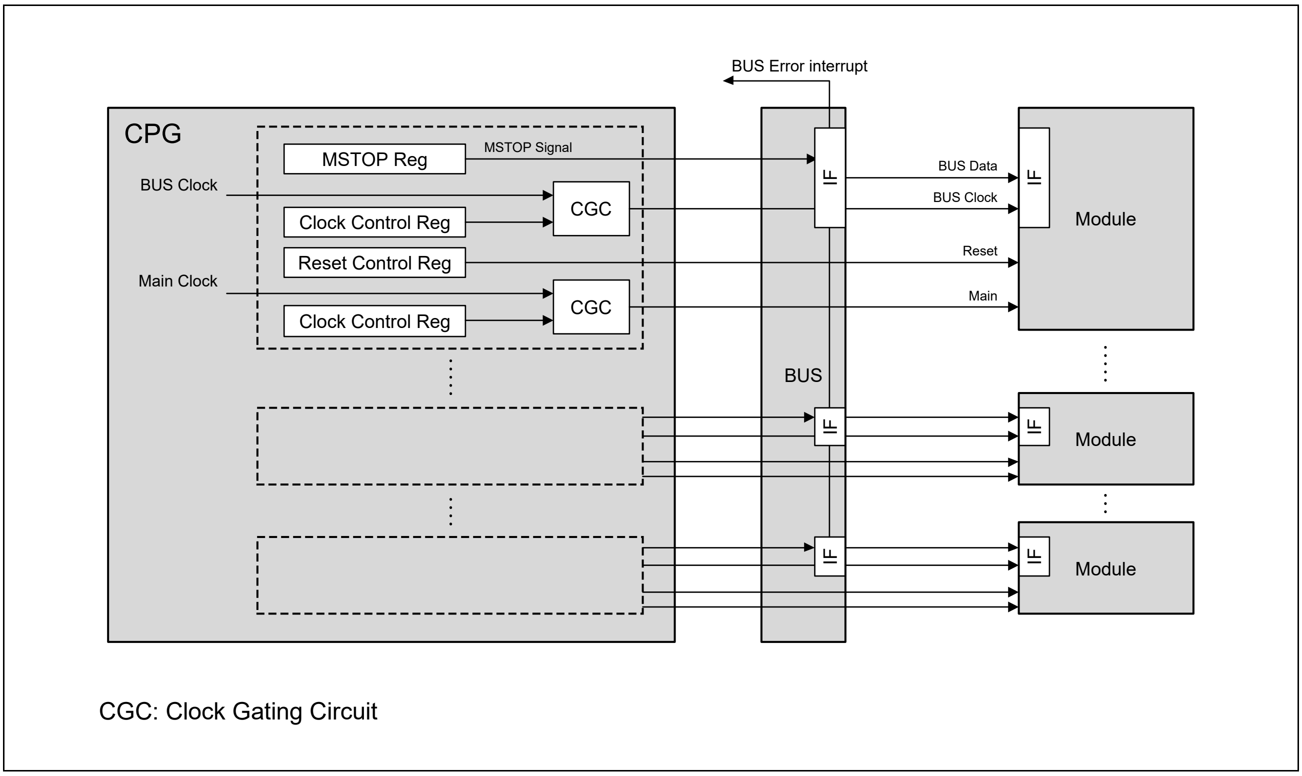


Figure 42.1 Block connection overview for Module Standby Mode

42.2.2 Operation

The following chart shows the procedure to enter and return from the Module Standby Mode.

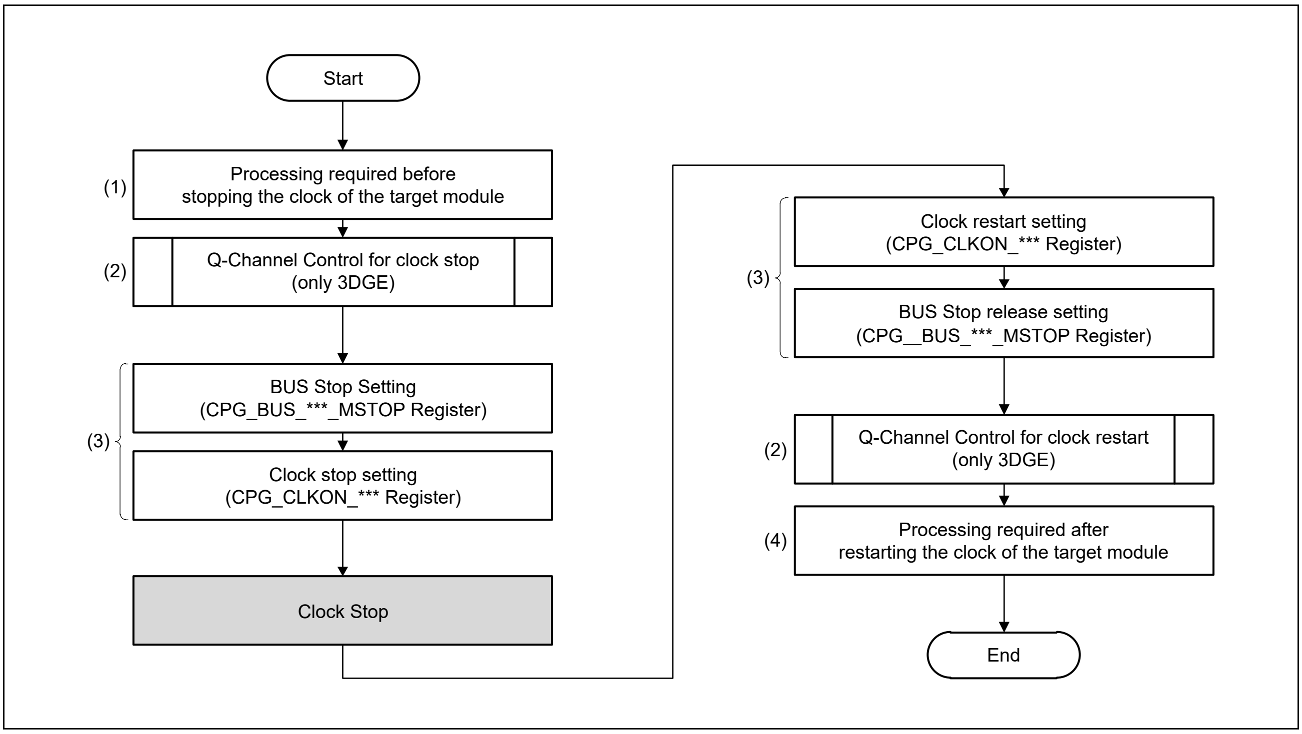


Figure 42.2 Module Standby Mode procedure

(1) Depend on the module, there may be a request to stop during communication, image processing, and various processing. In this case, if the managed CPU can be stopped unilaterally for each module, a stop request is immediately issued to the CPG. For those that cannot be stopped until a certain process is completed, after confirming that the managed CPU has stopped for each module, a stop request is made to the CPG.

(2) 3DGE requires Q-channel control before stopping and restarting the clock. Refer to **Section 42.2.2.2, Q-Channel control for 3DGE** regarding with the detail procedure.

(3) Refer to **Section 42.2.2.1, Setting of Clock control and MSTOP register** regarding to the setting of Clock Control Register and MSTOP Register.

(4) Whether the reset control is required depends on the module; check the specifications of the target module. Set the reset control register to 0 to enter the reset state. Set 1 for the release from the reset state.

42.2.2.1 Setting of Clock control and MSTOP register

The following table shows the combination of the Clock Control register and the MSTOP register to stop the clock of the target module.

The Module Standby Mode can be applied only to the modules shown in the table below.

- When entering the Module Standby Mode:

  Set the MSTOP Register bit of the target module to 1.
  Set the Clock Control Register bit of the target module to 0.

- When exiting the Module Standby Mode:

  Set the MSTOP Register bit of the target module to 0.
  Set the Clock Control Register bit of the target module to 1.

Table 42.3 Registers for Module Standby Mode (1/2)

Target module for Module Standby Mode	Clock Control Register		MSTOP Register	
	Register Name (CPG_CLKON_***)	bit	Register Name (CPG_BUS_***_MSTOP)	bit
GIC	GIC600	[0]	REG1	[7]
IA55	IA55	[1:0]	PERI_CPU	[13]
IM33	IM33	[1:0]	PERI_CPU	[14]
MHU	MHU	[0]	CPG_MHU_STOP*1	[0]
CoreSight	CST	[10:0]	PERI_CPU	[2]
SYC	SYC	[1]	PERI_CPU	[3]
DMAC	DMAC_REG	[1:0]	REG1	[3:0]
GTM ch0	GTM	[0]	REG0	[4]
GTM ch1		[1]		[5]
GTM ch2		[2]		[6]
MTU	MTU	[0]	MCPU1	[2]
POE3	POE3	[0]	MCPU1	[9]
GPT	GPT	[0]	MCPU1	[4]
POEGA	POEG	[0]	MCPU1	[5]
POEGB		[1]		[6]
POEGC		[2]		[7]
POEGD		[3]		[8]
WDT ch0	WDT	[1:0]	REG0	[2]
WDT ch1		[3:2]		[3]
WDT ch2		[5:4]		[1]
SPI Multi	SPI_MULTI	[1:0]	MCPU1	[1]
SDHI ch0	SDHI	[3:0]	PERI_COM	[0]
SDHI ch1		[7:4]		[1]
3DGE	GPU	[0]	REG1	[4]
ISU	ISU	[1:0]	PERI_VIDEO	[4]
VCPL4	H264	[0]	PERI_VIDEO	[2:0]
CRU	CRU	[3:0]	PERI_VIDEO	[3]
MIPI DSI	MIPI_DSI	[5:0]	PERI_VIDEO	[6:5]
LCDC	LCDC	[1:0]	PERI_VIDEO	[9:7]

Table 42.3 Registers for Module Standby Mode (2/2)

Target module for Module Standby Mode	Clock Control Register		MSTOP Register	
	Register Name (CPG_CLKON_***)	bit	Register Name (CPG_BUS_***_MSTOP)	bit
SSI ch0	SSI	[1:0]	MCPU1	[10]
SSI ch1		[3:2]		[11]
SSI ch2		[5:4]		[12]
SSI ch3		[7:6]		[13]
SRC	SRC	[0]	MCPU1	[3]
USB2.0 ch0	USB	[0], [2]	PERI_COM	[6:5]
USB2.0 ch1		[1]		[7]
USB2.0 PHY*2		[3]		[4]
ETHER ch0	ETH	[0]	PERI_COM	[2]
ETHER ch1		[1]		[3]
I2C ch0	I2C	[0]	MCPU2	[10]
I2C ch1		[1]		[11]
I2C ch2		[2]		[12]
I2C ch3		[3]		[13]
SCIF ch0	SCIF	[0]	MCPU2	[1]
SCIF ch1		[1]		[2]
SCIF ch2		[2]		[3]
SCIF ch3		[3]		[4]
SCIF ch4		[4]		[5]
SCI ch0	SCI	[0]	MCPU2	[7]
SCI ch1		[1]		[8]
IRDA	IRDA	[0]	MCPU2	[6]
RSPI ch0	RSPI	[0]	MCPU1	[14]
RSPI ch1		[1]		[15]
RSPI ch2		[2]	MCPU2	[0]
CANFD	CANFD	[0]	MCPU2	[9]
ADC	ADC	[1:0]	MCPU2	[14]
TSU	TSU	[0]	MCPU2	[15]

Note 1. CPG_MHU_STOP itself is the register name.

Note 2. The clock for USB2.0 PHY can only be stopped when the clock for USB2.0 ch0 and ch1 are stopped.

42.2.2.2 Q-Channel control for 3DGE

3DGE requires Q-Channel control before stopping and restarting the clock.

Refer to AMBA® Low Power Interface Arm® Q-Channel and P-Channel Interfaces regarding with the Q-Channel.

3DGE has five Q-Channels as following figure.

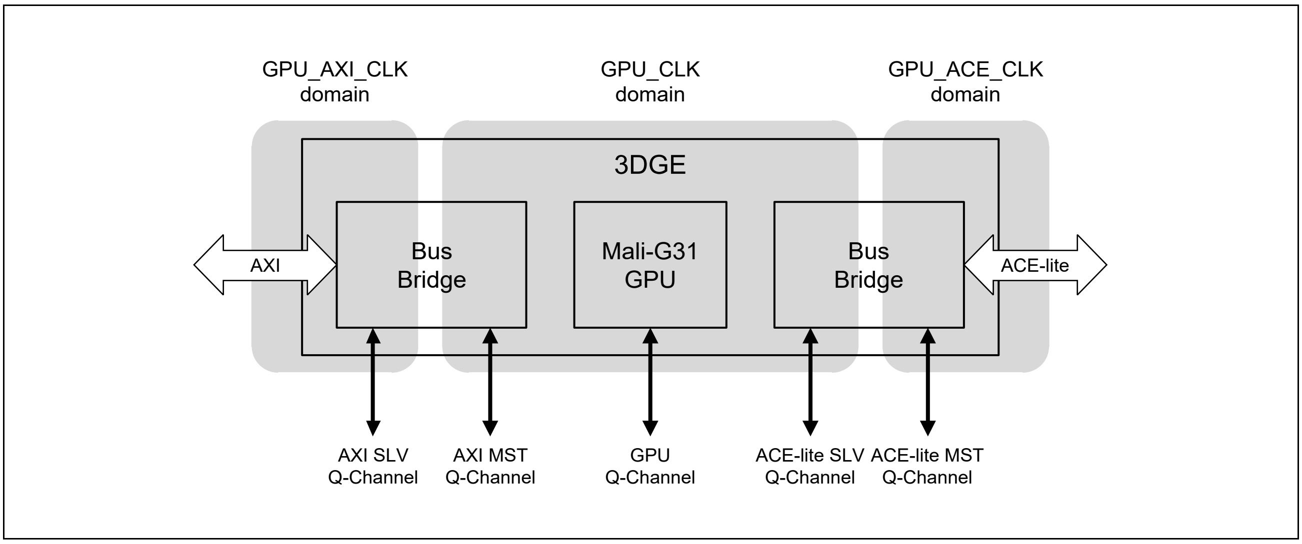


Figure 42.3 3DGE Q-Channel

Five Q-Channels are controlled by SYS_LP_GPU_CTL register.

Refer to **Section 6, System Controller (SYSC)** for detail.

The correspondence of SYS_LP_GPU_CTL bit and Q-Channel signal is below.

- SYS_LP_GPU_CTL[28:24] indicates the status of QDENY signals.
- SYS_LP_GPU_CTL[20:16] indicates the status of QACCEPTn signals.
- SYS_LP_GPU_CTL[12:8] controls QREQn signals.
- SYS_LP_GPU_CTL[4:0] indicates the status of QACTIVE signals.

Q-Channel Control for clock stop.

The procedure of Q-Channel control for clock stop is as follows.

(1) Confirm whether the QACTIVE signals of five Q-Channels are all 0.

(2) Set the QREQn signals of five Q-Channels to all 0.
Confirm the QACCEPTn signals and QDENY signals of five Q-Channels are all 0. Confirm whether the QACTIVE signal of AXI4 SLV Q-Channel is 0 again just in case. Now it is possible to stop clock. The only GPU CLK can be stopped.

(3) If the QACCEPTn signals and the QDENY signals of five Q-Channels are not all 0, that means it is impossible to stop clocks.
In this case, the QREQn signals of five Q-Channels are all returned to 1 to return to the step (1) again.

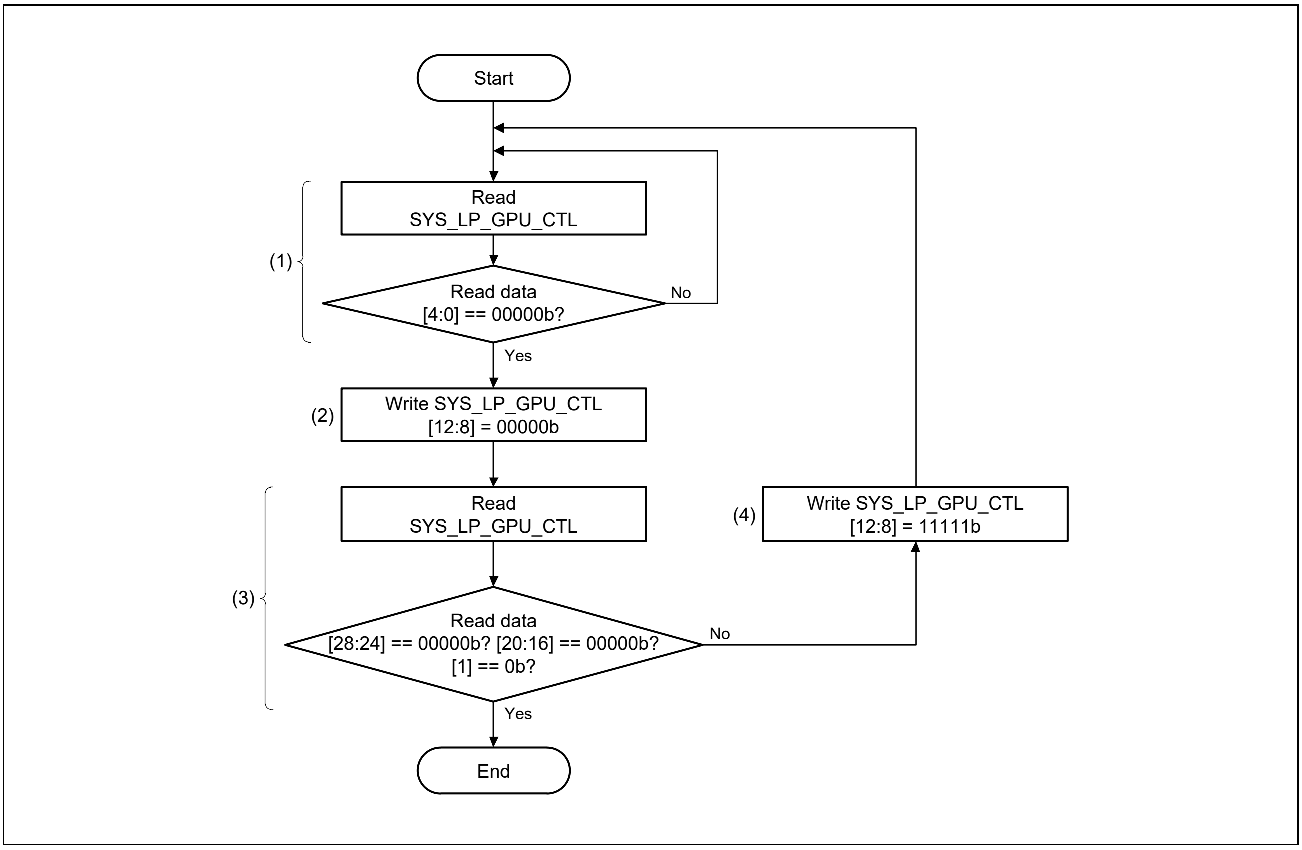


Figure 42.4 3DGE Q-Channel Control for clock stop

Q-Channel control for clock restart.

The procedure of Q-Channel control of clock is as follows.

(1) After the clock is restarted, return the QREQn signals of five Q-Channels to 1.

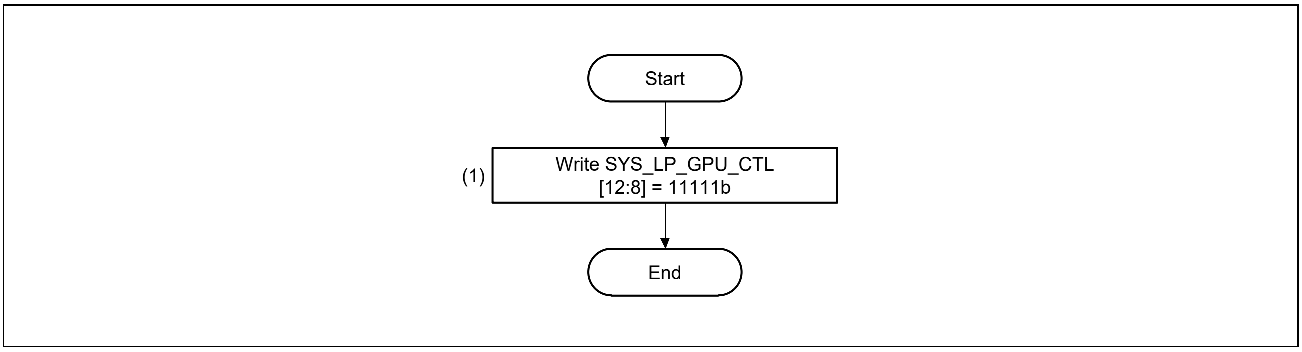


Figure 42.5 3DGE Q-Channel Control for clock restart

42.3 Cortex-M33 Sleep Mode

42.3.1 Overview

Refer to **Section 3.4.4, Cortex-M33 Sleep Mode** for detail.

The following table shows the registers related to the Cortex-M33 Sleep Mode.

Refer to **Section 6, System Controller (SYSC)** for detail.

Table 42.4 Cotrex-M33 Sleep Mode related registers

Register name	Abbreviation	Bit name
Lowpower Sequence Control Register1	SYS_LP_CTL1	bit[28]: CM33SLEEP_ACK bit[12]: CM33SLEEP_REQ
Lowpower Sequence Control Register5	SYS_LP_CTL5	bit[10]: CM33SLEEP_F
Lowpower Sequence Control Register6	SYS_LP_CTL6	bit[10]: CM33SLEEP_E
Lowpower Sequence Control Register7	SYS_LP_CTL7	bit[0]: IM33_MASK

42.3.2 Operation

This section shows an example procedure.

The following figure is one exmple how to use the SYSC registers. The SYS_LP_CTL1, the SYS_LP_CTL5 and the SYS_CTL6 registers can be used to share the transition state of Cortex-M33 with Cotrex-A55. Refering to these registers, Cortex-A55 can peform processing if need when Cortex-M33 enters and exits the Cortex-M33 Sleep Mode.

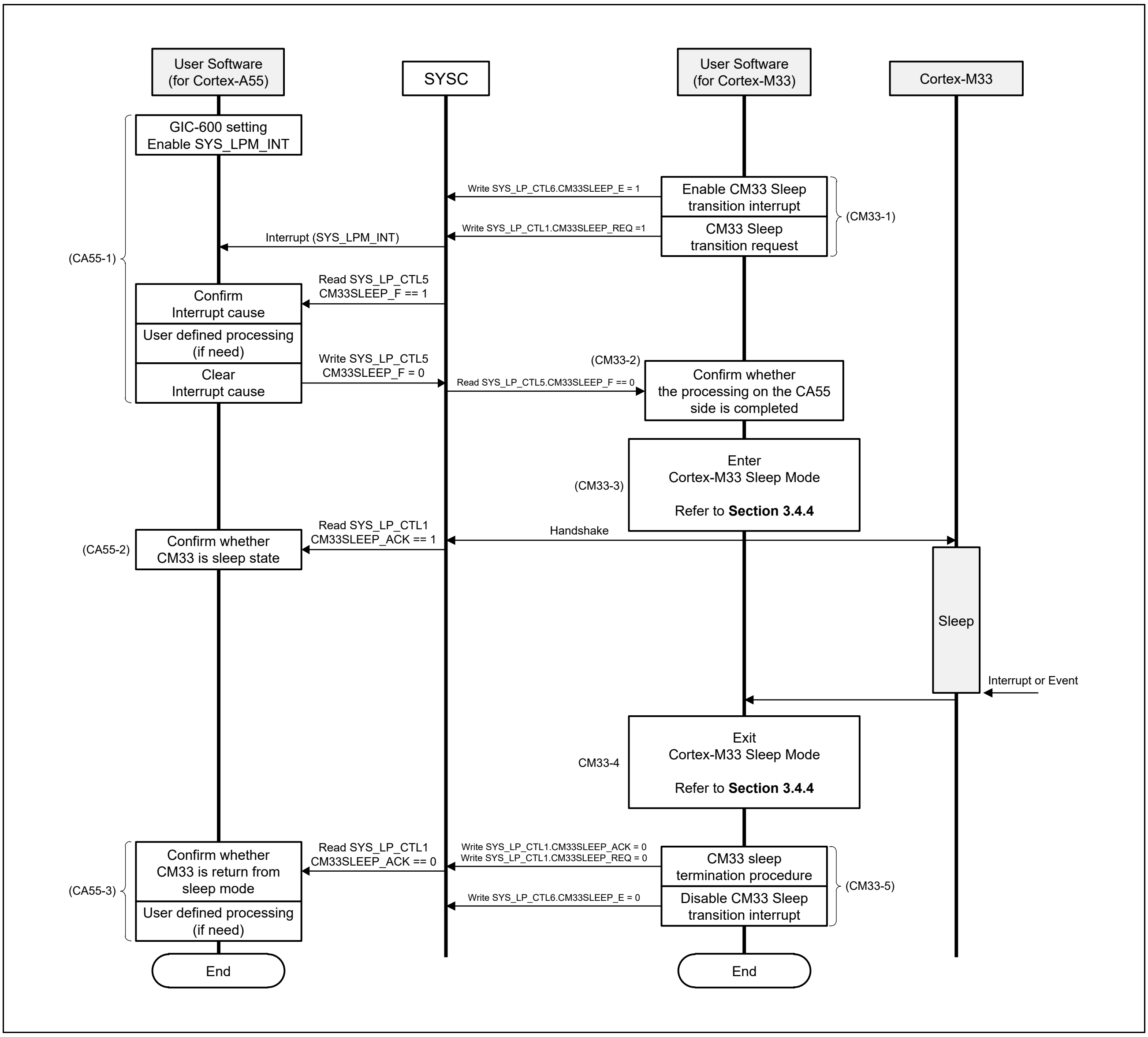


Figure 42.6 The example procedure how to use the SYSC registers (Cortex-M33 Sleep Mode)

(CM33-1) Set the CM33SLEEP_E bit of the SYS_LP_CTL6 register to 1 to enable the transition request interrupt to the Cortex-M33 Sleep Mode.
Setting the CM33SLEEP_REQ bit of the SYS_LP_CTL1 register to 1 causes a SYS_LPM_INT interrupt.
The CM33SLEEP_F bit of the SYS_LP_CTL5 register is set to 1 as an interrupt cause.

(CA55-1)	When a SYS_LPM_INT interrupt is detected, the SYS_LP_CTL5 register is read to check the interrupt cause. (Enable the SYS_LPM_INT interrupt on the GIC in advance.) By confirming that the CM33SLEEP_F bit is 1, the software on Cortex-A55 can know that Cortex-M33 is about to transition to Sleep Mode. Perform the process required when Cotrex-M33 is the sleep mode. (if need) And then clear the CM33SLEEP_F bit of the SYS_LP_CTL5 register.
(CM33-2)	Read the SYS_LP_CTL5 register. By confirming that the CM33SLEEP_F is cleared, the software on Cortex-M33 can confirm that the processing on Cortex-A55 side is completed.
(CM33-3)	Enter the Cortex-M33 Sleep Mode. Refer to **Section 3.4.4, Cortex-M33 Sleep Mode** procedure (1) to (7).
(CA55-2)	Read SYS_LP_CTL1 register. By confirming that the CM33SLEEP_ACK bit is 1, the software on Cortex-A55 can confirm that Cortex-M33 completed transiting to the sleep mode.
(CM33-4)	Exit the Cortex-M33 Sleep Mode. Refer to **Section 3.4.4, Cortex-M33 Sleep Mode** procedure (8) to (12).
(CM33-5)	Clear the CM33SLEEP_ACK bit of the SYS_LP_CTL1 register to indicate returning from the sleep mode. Clear the CM33SLEEP_REQ bit of the SYS_LP_CTL1 register to release the Cortex-M33 sleep mode transition request. Set the CM33SLEEP_E bit of the SYS_LP_CTL6 register to 0 to disable the transition request interrupt to the Cortex-M33 Sleep Mode.
(CA55-3)	Read SYS_LP_CTL1 register. By confirming that the CM33SLEEP_ACK bit is cleared, the software on Cortex-A55 can confirm that Cortex-M33 completed returning from the sleep mode. Perform the process required when Cotrex-M33 returned from the sleep mode. (if need)

42.4 Cortex-A55 Sleep Mode

42.4.1 Overview

Refer to **Section 2.3.1, Cotrex-A55 Sleep Mode** for detail.

The following table shows the registers related to the Cortex-A55 Sleep Mode.

Refer to **Section 6, System Controller (SYSC)** for detail.

Table 42.5 Cotrex-A55 Sleep Mode related registers

Register name	Abbreviation	Bit name
Lowpower Sequence Control Register1	SYS_LP_CTL1	bit[25:24]: CA55SLEEP_ACK[1:0] bit[9:8]: CA55SLEEP_REQ[1:0]
Lowpower Sequence Control Register2	SYS_LP_CTL2	bit[0]: CA55_STBYCTL
Lowpower Sequence Control Register5	SYS_LP_CTL5	bit[9]: CA55SLEEP1_F bit[8]: CA55SLEEP0_F
Lowpower Sequence Control Register6	SYS_LP_CTL6	bit[9]: CA55SLEEP1_E bit[8]: CA55SLEEP0_E

42.4.2 Operation

This section shows an example procedure.

The following figure is one exmple how to use the SYSC registers. The SYS_LP_CTL1, the SYS_LP_CTL5 and the SYS_CTL6 registers can be used to share the transition state of Cortex-A55 with Cotrex-M33. Refering to these registers, Cortex-M33 can peform processing if need when Cortex-A55 enters and exits the Cortex-A55 Sleep Mode.

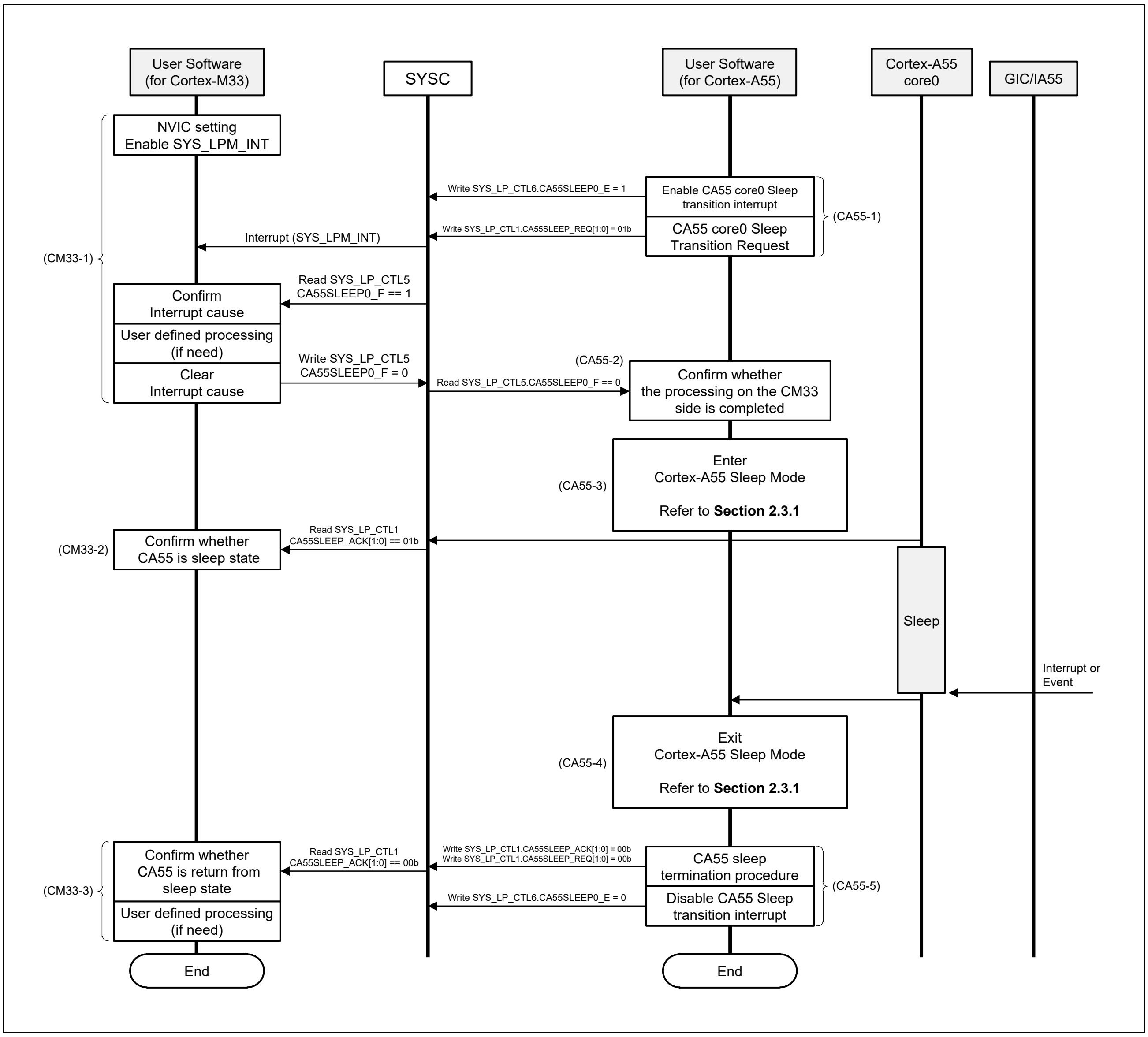


Figure 42.7 The example procedure how to use the SYSC registers (Cortex-A55 Sleep Mode)

(CA55-1) Set the CA55SLEEP0_E bit of the SYS_LP_CTL6 register to 1 to enable the transition request interrupt to the Cortex-A55 Sleep Mode (core0).
Setting the CA55SLEEP_REQ[1:0] bit of the SYS_LP_CTL1 register to 01b causes a SYS_LPM_INT interrupt.
The CA55SLEEP0_F bit of the SYS_LP_CTL5 register is set to 1 as an interrupt cause.

(CM33-1) When a SYS_LPM_INT interrupt is detected, the SYS_LP_CTL5 register is read to check the interrupt cause. (Enable the SYS_LPM_INT interrupt on the NVIC in advance.)
By confirming that the CA55SLEEP0_F bit is 1, the software on Cortex-M33 can know that Cortex-A55 core0 is about to transition to Sleep Mode.
Perform the process required when Cotrex-A55 is the sleep mode. (if need)
And then clear the CA55SLEEP0_F bit of the SYS_LP_CTL5 register.

(CA55-2) Read the SYS_LP_CTL5 register.
By confirming that the CA55SLEEP0_F is cleared, the software on Cortex-A55 can confirm that the processing on the Cortex-M33 side is completed.

(CA55-3) Enter the Cortex-A55 Sleep Mode.
Refer to **Section 2.3.1, Cortex-A55 Sleep Mode** procedure (1) to (3).

(CM33-2) Read SYS_LP_CTL1 register.
By confirming that the CA55SLEEP_ACK[1:0] bit is 01b, the software on Cortex-M33 can confirm that Cortex-A55 core0 completed transiting to the sleep mode.

(CA55-4) Exit the Cortex-A55 Sleep Mode.
Refer to **Section 2.3.1, Cortex-A55 Sleep Mode** procedure (4) to (6).

(CA55-5) Clear the CA55SLEEP_ACK[1:0] bit of the SYS_LP_CTL1 register to indicate returning from the sleep mode.
Clear the CA55SLEEP_REQ[1:0] bit of the SYS_LP_CTL1 register to release the Cortex-A55 sleep mode transition request.
Set the CA55SLEEP_E bit of the SYS_LP_CTL6 register to 0 to disable the transition request interrupt to the Cortex-CA55 Sleep Mode.

(CM33-3) Read SYS_LP_CTL1 register.
By confirming that the CA55SLEEP_ACK[1:0] bit is cleared, the software on Cortex-M33 can confirm that Cortex-A55 completed returning from the sleep mode.
Perform the process required when Cotrex-A55 returned from the sleep mode. (if need)

43. Debug Interface

43.1 Overview

This LSI has a debug interface for boundary scan function and debug support for Cortex-A55 and Cortex-M33.

43.1.1 Features

The functions of the debug interface of this LSI are shown below.

- Debug Interface
  - Support JTAG and SWD

- Debugger control function (DAP function)
  - Direct control of IP without going through CPU by switching Access Port (AP)

- Trace data support
  - Cortex-A55 trace output
  - Cortex-M33 trace output
  - Buffer function of trace data by ETF
  - Trace data output to system bus by using ETR

- Boundary Scan
  - Connection test between this LSI and other LSIs on the user board

- Other functions
  - Interlocking operation of Cortex-A55, Cortex-M33, system counter (SYC), watchdog timer, general timer (GTM), and debug component by cross trigger
  - Support WDT counter stop control function
  - Add time information to trace data by Time Stamp (24MHz operation)

43.1.2 Block Diagram

Figure 43.1 shows the block diagram of the debug interface, and **Figure 43.2** shows the block diagram of the Core Sight System (CST).

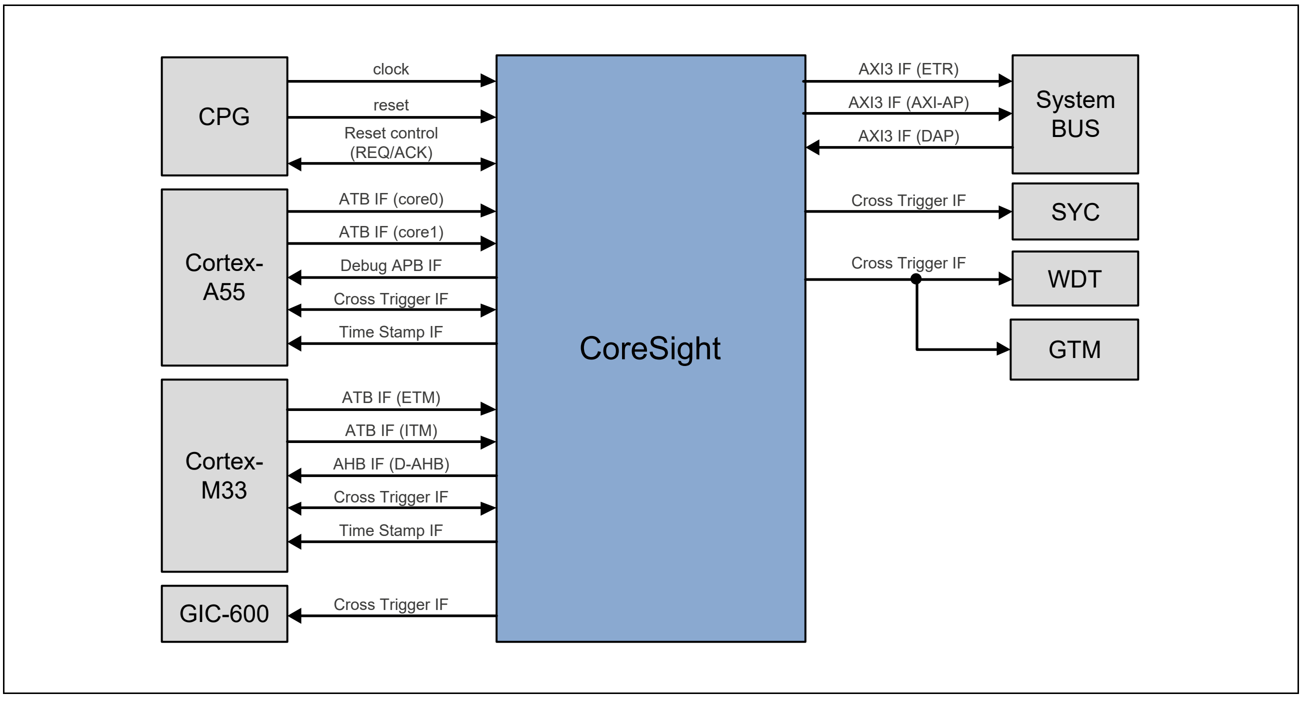


Figure 43.1 Debug System Block Diagram

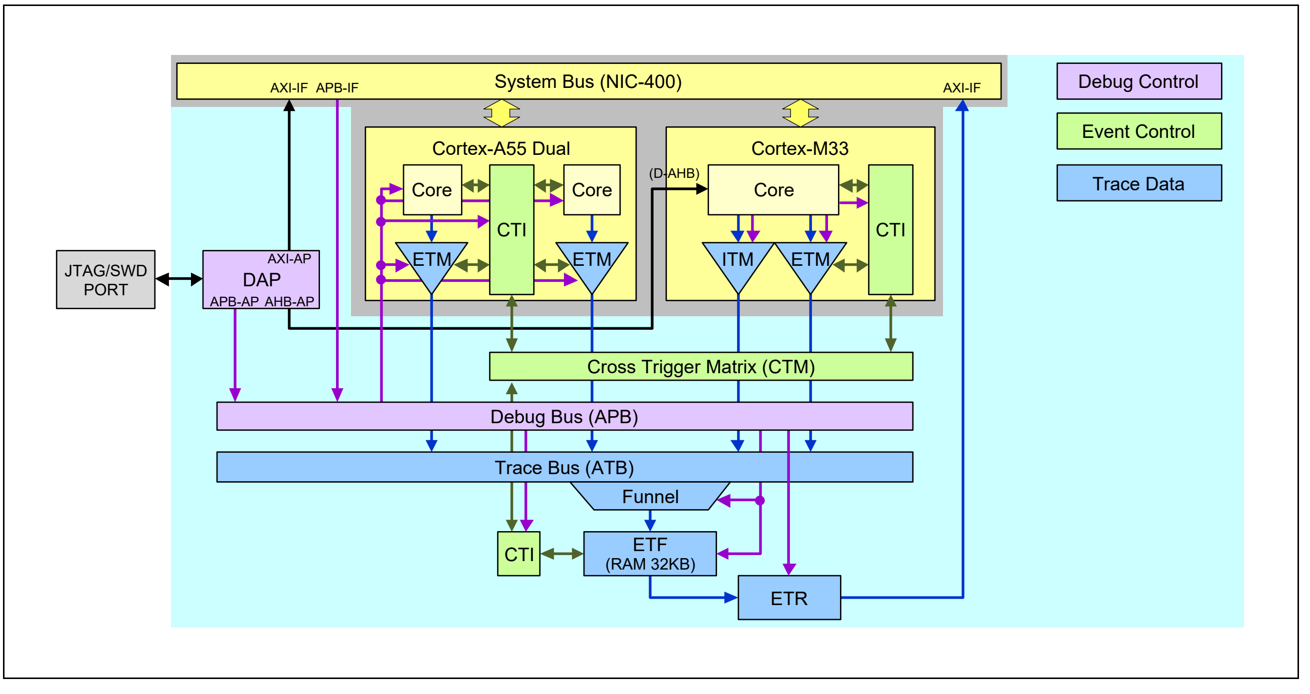


Figure 43.2 Core Sight System Block Diagram

- Debug Bus (APB)

  Bus used for controlling debug functions and acquiring debug data (register values and trace data in On-Chip Buffer)

  - DAP: Debug Access Port

- Cross Trigger Matrix (CTM)

  Functions required for co-debugging triggered by mutual events between CPU and Component

  - CTI: Cross Trigger Interface

- Trace Bus (ATB)

  A data bus which the trace data output by ETM flows

  - ETM: Embedder Trace Macrocell
  - ETR: Embedder Trace Router
  - ETF: Embedder Trace FIFO
  - Funnel: Trace data merge function

For details on ETM, ETR, and ETF, refer to “CoreSight Trace Memory Controller Technical Reference Manual”.

43.1.3 External Pins

- JTAG pins

The debug interface supports JTAG IF and SWD IF. **Table 43.1** shows the JTAG pins and their functions.

Table 43.1 JTAG Pins

Pin Name	I/O	Function		Name
		JTAG IF	SWD IF	
TCK/SWDCLK	Input	TCK	SWCLK	Test clock, Serial wire clock
TMS/SWDIO	Input, Input/Output	TMS	SWDIO	Test mode select, Serial wire data input/output
TDI	Input	TDI	—	Test data input
TDO	Output	TDO	—	Test data output
TRST#	Input	TRST	—	Test reset

- DEBUGEN pins

Table 43.2 shows the functions of the DEBUGEN pin.

Table 43.2 DEBUGEN Pins

Pin Name	I/O	Function
DEBUGEN	Input	Switches between normal operation and debug operation. 0: Normal operation 1: Debug operation

The DEBUGEN value must be fixed before releasing the power-on reset (PRST#). **Table 43.3** shows the difference in operation between normal operation and debug operation.

Table 43.3 Difference in Operation by DEBUGEN

	Normal Operation (DEBUGEN＝0)	Debug Operation (DEBUGEN＝1)	Related Module
Debug function module (CoreSight (other than DAP))	Boot impossible (Reset state)	Boot (Refer **Section 43.6.1** for reset condition and range)	CPG (Reset control)

- BSCANP pins

Table 43.4 shows the functions of the BSCANP pin.

Table 43.4 BSCANP Pins

Pin Name	I/O	Function
BSCANP	Input	Switches between normal operation and boundary scan test mode. 0: Normal operation 1: Boundary scan test mode

Boundary scan test mode is a mode in which a connection test is performed between the terminals of this LSI and other ICs connected on the customer's board and is performed when BSCANP = 1.

The actual access is controlled by the 5 terminals of JTAG when BSCANP = 1.

Table 43.5 shows the operation modes of this LSI according to the values of the DEBUGEN pin and BSCANP pin.

Table 43.5 Operating Mode

Pin Name		
BSCANP	DEBUGEN	Operating Mode
0	0	Normal operation
0	1	Debug operation
1	*	Boundary scan test mode

43.1.4 Treatment of Unused Pins

When the external terminal of the debug interface is not used, handle it as shown in **Table 43.6**.

Table 43.6 Handling of Unused Pins

Pin Name	Handle*[1]
TCK/SWDCLK	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level)
TMS/SWDIO	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level)
TDI	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level)
TDO	Open-circuit
TRST#*[2]	Fix this pin at a low level (pull down or connect to the ground level)
DEBUGEN	Fix this pin at a low level (pull down or connect to the ground level)
BSCANP	Fix this pin at a low level (pull down or connect to the ground level)

Note 1. Mount the pull-up/pull-down resistors on the board side

Note 2. TRST# should be fixed at low level in non-debug mode, and TRST# should be changed from low level to high level in debug mode.

43.2 Address Space

The CoreSight of this LSI has a 4 MB debug system address space. The address space of the debug system is shown in **Table 43.7**.

Table 43.7 Address Space of the Debug System

Address	Component*[1]
CoreSight Debug Components (Overall Address Space of this LSI)	
H'0_10C0_0000 ~ H'0_10C0_FFFF	CoreSight: ROM table
H'0_10C1_0000 ~ H'0_10C1_FFFF	CoreSight: Timestamp generator
H'0_10C2_0000 ~ H'0_10C2_FFFF	CoreSight: ETF
H'0_10C3_0000 ~ H'0_10C3_FFFF	CoreSight: ETR
H'0_10C4_0000 ~ H'0_10C4_FFFF	CoreSight: Trace Funnel
H'0_10C5_0000 ~ H'0_10C5_FFFF	CoreSight: CTI0
H'0_10C6_0000 ~ H'0_10C6_FFFF	CoreSight: CTI1
H'0_10C7_0000 ~ H'0_10CE_FFFF	CoreSight: Reserved
H'0_10CF_0000 ~ H'0_10CF_FFFF	CoreSight: Reserved
Reserved	
H'0_10D0_0000 ~ H'0_10DF_FFFF	Reserved
Cortex-A55 Debug Components (Overall Address Space of this LSI)	
H'0_10E0_0000 ~ H'0_10E0_FFFF	Cortex-A55: ROM table
H'0_10E1_0000 ~ H'0_10E1_FFFF	Cortex-A55: Core0 Debug
H'0_10E2_0000 ~ H'0_10E2_FFFF	Cortex-A55: Core0 CTI
H'0_10E3_0000 ~ H'0_10E3_FFFF	Cortex-A55: Core0 PMU
H'0_10E4_0000 ~ H'0_10E4_FFFF	Cortex-A55: Core0 ETM
H'0_10E5_0000 ~ H'0_10ED_FFFF	Reserved
H'0_10EE_0000 ~ H'0_10EE_FFFF	Cortex-A55: DSU CTI
H'0_10EF_0000 ~ H'0_10F0_FFFF	Reserved
H'0_10F1_0000 ~ H'0_10F1_FFFF	Cortex-A55: Core1 Debug
H'0_10F2_0000 ~ H'0_10F2_FFFF	Cortex-A55: Core1 CTI
H'0_10F3_0000 ~ H'0_10F3_FFFF	Cortex-A55: Core1 PMU
H'0_10F4_0000 ~ H'0_10F4_FFFF	Cortex-A55: Core1 ETM
H'0_10F5_0000 ~ H'0_10FF_FFFF	Cortex-A55: Reserved
Cortex-M33 Debug Components (Cortex-M33 Address Space)	
H'E000_0000 ~ H'E000_0FFF	Cortex-M33: ITM
H'E000_1000 ~ H'E000_1FFF	Cortex-M33: DWT
H'E000_2000 ~ H'E000_2FFF	Cortex-M33: FPB
H'E000_3000 ~ H'E000_DFFF	Cortex-M33: Reserved
H'E000_E000 ~ H'E000_EFFF	Cortex-M33: Secure SCS
H'E000_F000 ~ H'E002_DFFF	Cortex-M33: Reserved
H'E002_E000 ~ H'E002_EFFF	Cortex-M33: Non-Secure SCS
H'E002_F000 ~ H'E004_0FFF	Cortex-M33: Reserved
H'E004_1000 ~ H'E004_1FFF	Cortex-M33: ETM
H'E004_2000 ~ H'E004_2FFF	Cortex-M33: CTI
H'E004_3000 ~ H'E00F_EFFF	Cortex-M33: Reserved
H'E00F_F000 ~ H'E00F_FFFF	Cortex-M33: ROM Table

Note 1. Access to the Reserved area is prohibited. Operation is not guaranteed when accessed.

NOTE

- The CoreSight Debug Components area and Cortex-A55 Debug Components area can be accessed via the system bus and APB-AP.
  APB-AP is connected to port-0 of DAP. For the port configuration of DAP, refer to **Section 43.6.2, DAP**.
- The Cortex-M33 Debug Components area is accessible via AHB-AP.
  The AHB-AP for the area access is connected to port-2 of the DAP. For the port configuration of DAP, refer to **Section 43.6.2, DAP**.
- Each component is placed on a 64KB address boundary according to the ARMv8 Debug memory map. As an entity, the first 4 KB is valid.

43.3 Register Descriptions

43.3.1 ID Register (BSID)

The BSID register is a 32-bit register that cannot be accessed from the CPU. When the BSCANP pin is 1, it can be read from the TDO pin by setting the IDCODE command on the TAP controller. It cannot be written.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSID[31:16]															
Initial Value	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	1
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSID[15:0]															
Initial Value	1	1	0	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BSID[31:0]	H'1841_C447	—	This is the ID register specified in JTAG. The upper 4 bits (bit31-28) may change depending on the chip version.

43.3.2 Other Registers

For details on each register other than the BSID register, refer to "ARM CoreSight SoC-400 Technical Reference Manual" and "CoreSight Trace Memory Controller Technical Reference Manual".

43.4 Debug Connection

43.4.1 Debug Connection Mode

There are the following four modes for the debug connection method of this LSI.

Authentication Mode	Method
No authentication	Allow access from the debugger without authentication (initial setting)
SEC authentication*[1]	Allow access from the debugger after authentication is OK. (authentication method is security authentication by security IP)
Authentication refusal*[1]	Block access from the debugger

Note 1. SEC authentication and Authentication refusal can only be authenticated with secure products.

For SEC authentication modes, enter the authentication code from the debugger is needed.

For information on the authentication function of secure products, please contact our sales.

43.5 Debug Access Range

The debug access range by CoreSight is for all slaves as well as Cortex-A55. For details, refer to **Section 5.3, Accessible Areas**.

43.6 Operation

43.6.1 Debug Related Reset

Please refer to **Section 7.3.2.1, Range of Reset Application** for debug-related reset and reset range.

43.6.2 DAP

DAP is equipped with SWJ-DP as Debug Port (DP) and APB-AP, AXI-AP, AHB-AP as Access Port (AP).

The DAP port numbers are shown in **Table 43.8**. Indicates the correspondence between the port number specified in the DP SELECT register and the AP, and its usage.

Table 43.8 DAP Port Number

Port number	AP	Usage
Port-0	APB-AP	Debug Component recognition and settings
Port-1	AXI-AP	System IP control (access to system bus)
Port-2	AHB-AP	Cortex-M33 settings

43.6.3 Trace

43.6.3.1 Data Flow

In CoreSight, all trace data is aggregated once by Trace Funnel, and stored in ETF, or output to the system bus via ETR in the subsequent stage. Setting this port number to the Trace Funnel according to the route of the trace data is needed. The Trace Funnel port numbers are shown in **Table 43.9**.

Table 43.9 Trace Funnel Port Number

Port Number	Source Components
Port-0	CA55: Core0 ETM
Port-1	CA55: Core1 ETM
Port-2	CM33: ITM
Port-3	CM33: ETM

43.6.3.2 Timestamp

If time information want to be included in the trace data, the Timestamp Generator can be used. The timestamp replicator port numbers are shown in **Table 43.10**. The Timestamp Generator runs on a 24-MHz clock.

Table 43.10 Timestamp Replicator Port Number

Port Number	Target Components
Port-0	CA55: Core0/1 ETM
Port-1	CM33

43.6.4 Cross Trigger

The CTI (Cross Trigger Interface) implemented in CoreSight, Cortex-A55, and Cortex-M33 is used to communicate debug events.

43.6.4.1 CoreSight Cross Trigger Connectiion

CoreSight has two CTIs. The connection specifications are shown in **Table 43.11** to **Table 43.14**.

Table 43.11 CoreSight CTI0 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	(Unused)	—
[4]	(Unused)	—
[3]	ETR	ETR Event Output 1 (FULL)
[2]	ETR	ETR Event Output 0 (ACQCOMP)
[1]	ETF	ETF Event Output 1 (FULL)
[0]	ETF	ETF Event Output 0 (ACQCOMP)

Table 43.12 CoreSight CTI0 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	SYC	SYC Event input1 (RESTART REQ/ACK)
[6]	SYC	SYC Event input0 (HALT REQ)
[5]	WDT/OSTM	WDT/OSTM Event input1 (RESTART REQ/ACK)
[4]	WDT/OSTM	WDT/OSTM Event input0 (HALT REQ)
[3]	ETR	ETR Event Input 1 (TRIGIN)
[2]	ETR	ETR Event Input 0 (FLUSHIN)
[1]	ETF	ETF Event Input 1 (TRIGIN)
[0]	ETF	ETF Event Input 0 (FLUSHIN)

Table 43.13 CoreSight CTI1 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	(Unused)	—
[4]	(Unused)	—
[3]	(Unused)	—
[2]	(Unused)	—
[1]	(Unused)	—
[0]	(Unused)	—

Table 43.14 CoreSight CTI1 trigger output

Trigger Output Bit	Destination Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	(Unused)	—
[4]	(Unused)	—
[3]	(Unused)	—
[2]	(Unused)	—
[1]	Cortex-A55	Cortex-A55 Event Input (PMUSNAPSHOTREQ/PMUSNAPSHOTACK)
[0]	GIC-600	GIC-600 Event Input (sample_req/sample_ack)

43.6.4.2 Cortex-A55 Cross Trigger Connectiion

The Cortex-A55 has two CTIs (Cortex-A55 Core0 CTI and Cortex-A55 Core1 CTI). The connection specifications for each are shown in **Table 43.15** to **Table 43.18**.

Table 43.15 Cortex-A55 CTI0 trigger input

Trigger Input Bit	Source Components	Description
[7]	Cortex-A55: Core0 ETM	ETM Trace Output 3 trigger event
[6]	Cortex-A55: Core0 ETM	ETM Trace Output 2 trigger event
[5]	Cortex-A55: Core0 ETM	ETM Trace Output 1 trigger event
[4]	Cortex-A55: Core0 ETM	ETM Trace Output 0 trigger event
[3]	(Unused)	—
[2]	Cortex-A55: Core0 PE	Profiling sample trigger event
[1]	Cortex-A55: Core0 PE	Performance Monitors Overflow trigger event
[0]	Cortex-A55: Core0 PE	Cross-halt trigger event

Table 43.16 Cortex-A55 CTI0 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	Cortex-A55: Core0 ETM	Generic Trace External Input 3 trigger event
[6]	Cortex-A55: Core0 ETM	Generic Trace External Input 2 trigger event
[5]	Cortex-A55: Core0 ETM	Generic Trace External Input 1 trigger event
[4]	Cortex-A55: Core0 ETM	Generic Trace External Input 0 trigger event
[3]	(Unused)	—
[2]	External (GIC-600)	Generic CTI Interrupt trigger event
[1]	Cortex-A55: Core0 PE	Restart Request trigger event
[0]	Cortex-A55: Core0 PE	Debug Request trigger event

Table 43.17 Cortex-A55 CTI1 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	Cortex-A55: Core1 ETM	ETM Trace Output 3 trigger event
[6]	Cortex-A55: Core1 ETM	ETM Trace Output 2 trigger event
[5]	Cortex-A55: Core1 ETM	ETM Trace Output 1 trigger event
[4]	Cortex-A55: Core1 ETM	ETM Trace Output 0 trigger event
[3]	(Unused)	—
[2]	Cortex-A55: Core1 PE	Profiling sample trigger event
[1]	Cortex-A55: Core1 PE	Performance Monitors Overflow trigger event
[0]	Cortex-A55: Core1 PE	Cross-halt trigger event

Table 43.18 Cortex-A55 CTI1 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	Cortex-A55: Core1 ETM	Generic Trace External Input 3 trigger event
[6]	Cortex-A55: Core1 ETM	Generic Trace External Input 2 trigger event
[5]	Cortex-A55: Core1 ETM	Generic Trace External Input 1 trigger event
[4]	Cortex-A55: Core1 ETM	Generic Trace External Input 0 trigger event
[3]	(Unused)	—
[2]	External (GIC-600)	Generic CTI Interrupt trigger event
[1]	Cortex-A55: Core1 PE	Restart Request trigger event
[0]	Cortex-A55: Core1 PE	Debug Request trigger event

43.6.4.3 Cortex-M335 Cross Trigger Connectiion

The Cortex-M33 has one CTI (Cortex-M33 CTI). The connection specifications are shown in **Table 43.19** and **Table 43.20**.

Table 43.19 Cortex-M33 CTI Trigger Input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	Cortex-M33: ETM	ETM Event Output 1
[4]	Cortex-M33: ETM/Processor	ETM Event Output 0 or Comparator Output 3
[3]	Cortex-M33: Processor	DWT Comparator Output 2
[2]	Cortex-M33: Processor	DWT Comparator Output 1
[1]	Cortex-M33: Processor	DWT Comparator Output 0
[0]	Cortex-M33: Processor	Processor Halted

Table 43.20 Cortex-M33 CTI Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	Cortex-M33: ETM	ETM Event Input 3
[6]	Cortex-M33: ETM	ETM Event Input 2
[5]	Cortex-M33: ETM	ETM Event Input 1
[4]	Cortex-M33: ETM	ETM Event Input 0
[3]	Cortex-M33: System	Interrupt request 1
[2]	Cortex-M33: System	Interrupt request 0
[1]	Cortex-M33: Processor	Processor Restart
[0]	Cortex-M33: Processor	Processor debug request

43.6.5 WDT Counter Stop Control

This LSI has a WDT counter stop control function linked with a debug event to prevent an unintended reset by WDT during debug operation. When a debug event occurs, CoreSight asserts the WDT counter stop request signal, controls the WDT counter stop signal (CNTSTOP), and stops counting. The target WDT is all WDT. For details on the WDT count stop function, refer to **Section 6.4.2, WDT Stop control function**.

43.6.6 GTM Counter Stop Control

This LSI has a GTM counter stop control function linked with debug events to prevent unintended GTM counts during debug operation. When a debug event occurs, CoreSight asserts a GTM counter stop request signal to stop the GTM count.

43.6.7 SYC Counter Stop Control

This LSI has a SYC counter stop control function linked with debug events to prevent unintended SYC counts during debug operation. When a debug event occurs, CoreSight asserts a SYC counter stop request signal to stop the SYC count.

44. Thermal Sensor Unit (TSU)

This LSI chip incorporates a thermal sensor unit (TSU) that measures the temperature inside the LSI.

44.1 Overview

44.1.1 Features

- This LSI has a single TSU.
- The thermal sensor in this unit measures temperatures in the range from −40°C to 125°C with an accuracy of ±3°C.
- The TSU repeats measurement at 20-µs intervals, and automatically updates the results of measurement.

44.1.2 Block Diagram

Figure 44.1 is a block diagram of the TSU.

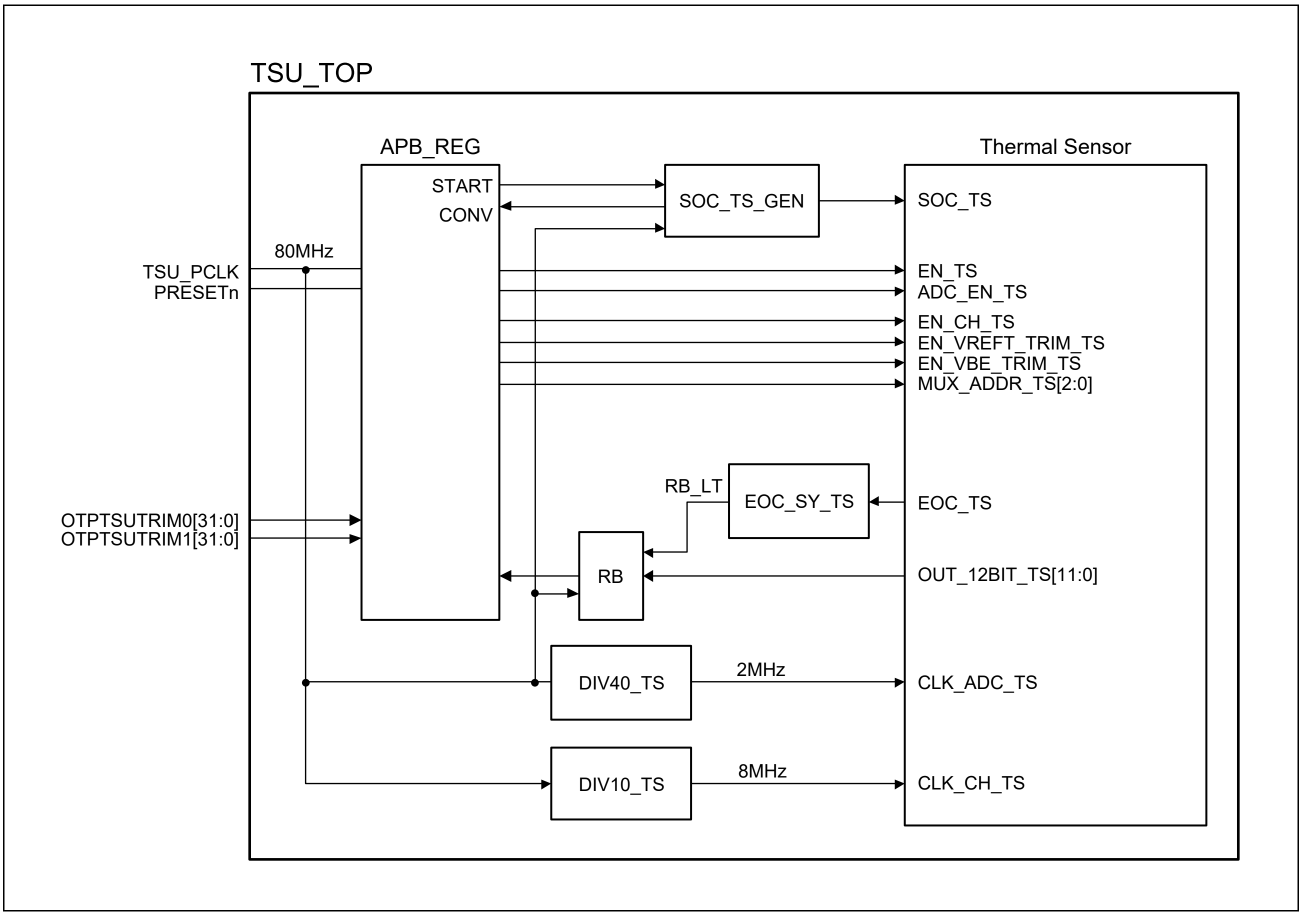


Figure 44.1 Block Diagram of the TSU

The TSU consists of the following blocks.

- APB_REG
- Thermal Sensor
- DIV40_TS and DIV10_TS
- SOC_TS_GEN
- EOC_SY_TS
- Read Buffer (RB)

44.1.2.1 APB_REG

- The APB_REG block provides an external interface for the TSU.
- This block contains control and status registers for the thermal sensor.

44.1.2.2 Thermal Sensor

- The thermal sensor block includes the internal probe, A/D converter and other items.
- The thermal sensor generates 12-bit digital values corresponding to the temperature and outputs them through the OUT_12BIT_TS[11:0] port.

44.1.2.3 DIV40_TS and DIV10_TS

- The TSU operates with the TSU_PCLK clock (80 MHz).
- The DIV40_TS block is a divide-by-40 divider which divides TSU_PCLK to generate a 2-MHz clock frequency to be supplied to the A/D converter in the thermal sensor as CLK_ADC_TS.
- The DIV10_TS block is a divide-by-10 divider which divides TSU_PCLK to generate an 8-MHz clock frequency to be supplied to the thermal sensor core as CLK_CH_TS.

44.1.2.4 SOC_TS_GEN

- The SOC_TS_GEN block generates the SOC_TS signal to be supplied to the thermal sensor for control over the timing of its operations.
- The SOC_TS_GEN block generates the SOC_TS signal every 20 µs. This is triggered by the START signal from the APB_REG block.
- The SOC_TS_GEN block generates the CONV signal that indicates the operating state of the thermal sensor. The value of the CONV signal can be read through the APB_REG block.

44.1.2.5 EOC_SY_TS

- The EOC_SY_TS block detects the EOC_TS signal from the thermal sensor and generates the RB_LT signal in synchronization with TSU_PCLK.
- The EOC_TS signal indicates the end of A/D conversion of the voltage for the temperature measured by the thermal sensor.

44.1.2.6 Read Buffer (RB)

- The RB block captures the 12-bit digital value from the OUT_12BIT_TS[11:0] port terminals of the thermal sensor in a register. This is triggered by the RB_LT signal from the EOC_SY_TS block. This value corresponds to the measured temperature.
- The value captured in the RB block can be read from the TSU through the APB_REG block.

44.1.3 External Pins

The TSU has no external pins.

44.1.4 Interrupts

The TSU has no interrupts.

44.2 Register Configuration

This section describes the registers for use in controlling the TSU.

Base Address: H’0_1005_9400 (Cortex-A55 Address Space)
Base Address: H’4005_9400 (Cortex-M33 Address Space Non-Secure)
Base Address: H’5005_9400 (Cortex-M33 Address Space Secure)

44.2.1 List of Register

Table 46 1 lists the TSU registers. All registers are accessible in 32-bit units.

Table 44.1 List of the TSU Registers

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Sensor Mode Register	TSU_SM	R/W	H’0000_0000	H’00	32
Sensor Trigger Register	TSU_ST	R/W	H’0000_0000	H’04	32
Sensor ADC Data Register	TSU_SAD	R	H’0000_0000	H’0C	32
Sensor Status Register	TSU_SS	R	H’0000_0000	H’10	32
TSU OTP Calibration Register 0	OTPTSUTRIM0_REG	R	—	H’18	32
TSU OTP Calibration Register 1	OTPTSUTRIM1_REG	R	—	H’1C	32

44.3 Register Descriptions

44.3.1 Sensor Mode Register (TSU_SM)

This register is used to specify the operating mode of the thermal sensor.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADC_EN_TS	ADC_EN_TS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	ADC_EN_TS	0b	R/W	These bits specify the operating mode of the thermal sensor.
0	EN_TS	0b	R/W	

ADC_EN_TS	EN_TS	Operating mode of the thermal sensor
0b	0b	Power-saving mode
0b	1b	ADC power-saving mode
1b	0b	Power-saving mode
1b	1b	Normal operating mode

44.3.2 Sensor Trigger Register (TSU_ST)

This register is used to start and stop the cycles of A/D conversion by the thermal sensor. The A/D conversion is repeated at 20-µs intervals once the conversion is started.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RSV	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	RSV	0b	R/W	The software may use this bit as a note. The written value is read.
0	START	0b	R/W	This bit starts and stops the cycles of A/D conversion by the thermal sensor. 0b: Stops the A/D conversion*[2] 1b: Starts the A/D conversion*[1]

Note 1. Before setting the START bit to 1b, be sure to set the TSU_SM register. Specifically, set the TSU_SM register, wait for 60 µs, and then set the START bit to 1b.

Note 2. When the START bit is to be set to 0b, be sure to read the A/D-converted value before making the setting.

44.3.3 Sensor ADC Data Register (TSU_SAD)

This register holds the digital value after A/D conversion of the voltage from the thermal sensor.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	OUT_12BIT_TS											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 0	OUT_12BIT_TS	H'000	R	These bits hold the digital value after A/D conversion of the voltage from the thermal sensor. The value is 12-bit unsigned.

44.3.4 Sensor Status Register (TSU_SS)

This register indicates the state of A/D conversion in the thermal sensor.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CONV
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	CONV	0b	R	These bits indicate the state of A/D conversion in the thermal sensor. 0b: A/D conversion is stopped. 1b: A/D conversion is ongoing.

44.3.5 TSU OTP Calibration Register 0 (OTPTSUTRIM0_REG)

This register holds the correction value for calibration to be used in calculating the temperature

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTPTSUTRIM0_EN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	OTPTSUTRIM0											
Initial Value	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	OTPTSUTRIM0_EN	—	R	This bit indicates whether [11:0] OTPTSUTRIM0 is valid or invalid. 0b: [11:0] OTPTSUTRIM0 is invalid. 1b: [11:0] OTPTSUTRIM0 is valid.
30 to 12	—	—	R	Reserved When read, the value is indefinite value. The written value will be ignored.
11 to 0	OTPTSUTRIM0	—	R	These bits hold the correction value for calibration to be used in calculating the temperature.

44.3.6 TSU OTP Calibration Register 1 (OTPTSUTRIM1_REG)

This register holds the correction value for calibration to be used in calculating the temperature.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTPTSUTRIM1_EN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	OTPTSUTRIM1											
Initial Value	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	OTPTSUTRIM1_EN	—	R	This bit indicates whether [11:0] OTPTSUTRIM1 is valid or invalid. 0b: [11:0] OTPTSUTRIM1 is invalid. 1b: [11:0] OTPTSUTRIM1 is valid.
30 to 12	—	—	R	Reserved When read, the value is indefinite value. The written value will be ignored.
11 to 0	OTPTSUTRIM1	—	R	These bits hold the correction value for calibration to be used in calculating the temperature.

44.4 Operation

44.4.1 Procedure for Starting the A/D Conversion

This section describes the procedure for starting the A/D conversion. Follow this procedure to acquire the A/D converted values for the measured temperatures.

1. Start supplying the clock to the TSU.
2. Release the TSU from the reset state.
3. Set the EN_TS and ADC_EN_TS bits in the TSU_SM register to 1b to place the thermal sensor in the normal operating mode.
4. Wait for 60 µs, leaving the value of the START bit as 0b.
5. Set the START bit in the TSU_ST register to 1b. The thermal sensor then starts the cycles of A/D conversion and the CONV bit in the TSU_SS register indicates 1b.
6. Once 20 µs have passed, the first A/D converted value from the thermal sensor is automatically stored in the TSU_SAD register. The value in that register is then updated every 20 µs. Reading the TSU_SAD register returns the latest A/D converted value.

CAUTION

Ensure that the setting of START is 1b when reading an A/D converted value from the TSU_SAD register.

44.4.2 Procedure for Stopping the A/D Conversion

This section describes the procedure for stopping the A/D conversion.

1. Set the START bit in the TSU_ST register to 0.
2. Poll the CONV bit in the TSU_SS register until it indicates 0b (A/D conversion has stopped).
3. Once you have confirmed that conversion has stopped, set the EN_TS and ADC_EN_TS bits in the TSU_SM register to 0b to place the thermal sensor in the power-saving mode.

CAUTION

If you intend to restart the A/D conversion after having stopped it, start from step 3 in **Section 44.4.1, Procedure for Starting the A/D Conversion**.

44.4.3 Confirming the State of the A/D Conversion by Monitoring the CONV Bit

The A/D conversion by the TSU can be started and stopped by the START bit in the TSU_ST register. However, the timing of stopping the A/D conversion by the thermal sensor is not synchronous with the START bit being set to 0 (stopping the A/D conversion). Monitor the CONV bit in the TSU_SS register to confirm that the A/D conversion by the thermal sensor has stopped.

44.4.4 Procedure for Measuring the Temperature

To calculate the temperature Tj, read registers TSU_SAD, OTPTSUTRIM0_REG, and OTPTSUTRIM1_REG, and follow the procedure below.

1. Read the value of the TSU_SAD register eight times every at least 20 µs. The read values are defined as TSCode[0] to TSCode[7].

2. Calculate the average of the values read, TSCodeAVE, by using the following formula.

$$\mathrm{TSCodeAVE} = \frac{(\mathrm{TSCode}[0]\ +\ \cdots\ +\ \mathrm{TSCode}[7])}{8}$$

3. Apply curvature correction and calculate DSENSOR by using the following formula.

$$\mathrm{DSENSOR} = \frac{\mathrm{TSCodeAVE}}{(1\ +\ \mathrm{TSCodeAVE}\ \times\ 0.000013)}$$

4. Read the correction values for use in calibration from registers OTPTSUTRIM0_REG and OTPTSUTRIM1_REG and use the following formula to calculate Tj.

$$\mathrm{Tj} = (\mathrm{DSENSOR}\ -\ \mathrm{OTPTSUTRIM1}) \times \left(\frac{165}{(\mathrm{OTPTSUTRIM0}\ -\ \mathrm{OTPTSUTRIM1})}\right) - 40$$

44.5 Usage Note

When the START bit in the TSU_ST register is to be set to 0b, be sure to read the A/D-converted value in the TSU_SAD register before making the setting.

45. Trusted Secure IP

This LSI incorporates a Trusted Secure IP module to provide security functions. The module consists of an access management circuit, encryption engine, and random number generator. In combination with the Trusted Secure IP driver, the Trusted Secure IP can prevent eavesdropping (confidentiality), falsification of information (integrity), and impersonation (authenticity).

Key information to be used in encrypting and decrypting data is only stored within the Trusted Secure IP, and any external access can be shut out to obtain a system with strong security.

45.1 Overview

Table 45.1 summarizes the specifications of the Trusted Secure IP. **Figure 45.1** shows a block diagram of the Trusted Secure IP.

Table 45.1 Specifications of Trusted Secure IP (1/2)

Item	Description
Access control	Access management circuit ● In case of irregular access to the Trusted Secure IP due to a falsified program or runaway execution of a program, this circuit blocks all subsequent access and stops the output of data from the Trusted Secure IP.
Encryption engine	AES: Compliant with NIST FIPS PUB 197 algorithm ● Key sizes: 128, 192, or 256 bits ● Block sizes: 128 bits ● Block cipher mode of operation – ECB, CBC, CTR: Compliant with NIST SP 800-38A – CMAC: Compliant with NIST SP 800-38B – CCM: Compliant with NIST SP 800-38C – GCM: Compliant with NIST SP 800-38D – XTS: Compliant with NIST SP 800-38E – GCTR ● Number of cycles for execution*1 – ECB, CBC, CTR, CMAC, GCTR, XTS: 11 cycles of P1ϕ for 128-bit keys, 13 cycles of P1ϕ for 192-bit keys, 15 cycles of P1ϕ for 256-bit keys – CCM: 22 cycles of P1ϕ for 128-bit keys, 26 cycles of P1ϕ for 192-bit keys, 30 cycles of P1ϕ for 256-bit keys AES-GCM AES-GCM is realized by combining AES-GCTR and GHASH.

Table 45.1 Specifications of Trusted Secure IP (2/2)

Item	Description
Encryption engine	RSA ● Key sizes: Up to 4096 bits ● Block sizes: Up to 4096 bits ● Number of cycles for execution*[1] – Approximately 64,000 cycles (Encryption) / 13,000,000 cycles (Decryption) of P1ϕ. HASH ● Support for SHA1, SHA224/SHA256, GHASH ● Block sizes: 512 bits ● Number of cycles for execution*[1] – SHA1: 80 cycles of P1ϕ – SHA224/SHA256: 64 cycles of P1ϕ – GHASH: 9 cycles of P1ϕ ECC ● Compatible with ECDSA and ECDH ● Data block length: 256 bits
Encryption engine	Key management ● Keys are only valid within the Trusted Secure IP. ● Only key generation information is output from the Trusted Secure IP. ● Keys can be regenerated by the input of key generation information to the Trusted Secure IP. Endian ● Big or little
Generation of random numbers	32-bit true random number generator ● The Trusted Secure IP driver can assemble 32-bit true random numbers to generate 128- or 256-bit true random numbers. ● The generated 128-bit and 256-bit true random numbers are used as keys in encrypting and decrypting data.
Unique ID	● An ID unique to the LSI (unique ID) is accessible from the access management circuit through the dedicated bus. ● Combining the unique ID with the key generation information prevents the illicit copying of data to another LSI.
Interrupt sources	10
Low power consumption	Setting of the module stop state is possible.

Note 1. This does not include the overhead for calling functions of the Trusted Secure IP driver.

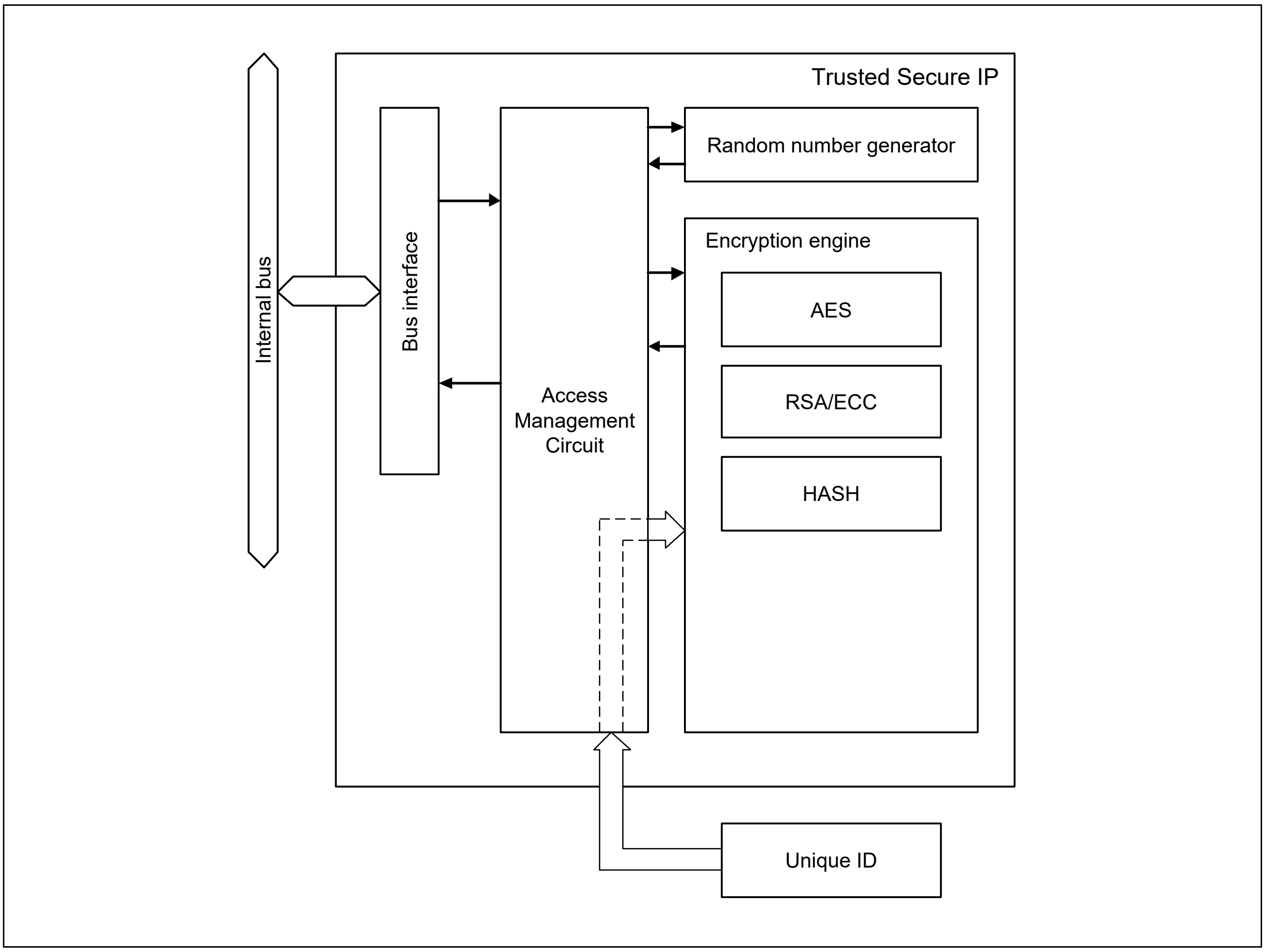


Figure 45.1 Trusted Secure IP Block Diagram

45.2 Operation

45.2.1 Operating Modes and State Transitions

Figure 45.2 shows the state transitions of the Trusted Secure IP.

Use of the Trusted Secure IP security functions is only possible through use of the Trusted Secure IP driver provided by Renesas Electronics, in accordance with the state transitions as shown in the figure below.

When irregular access to the Trusted Secure IP (access that violates the defined procedure) due to a falsified program or a program entering runaway execution, etc. is attempted, the access management circuit does not accept any subsequent access and stops the output of any data from the Trusted Secure IP.

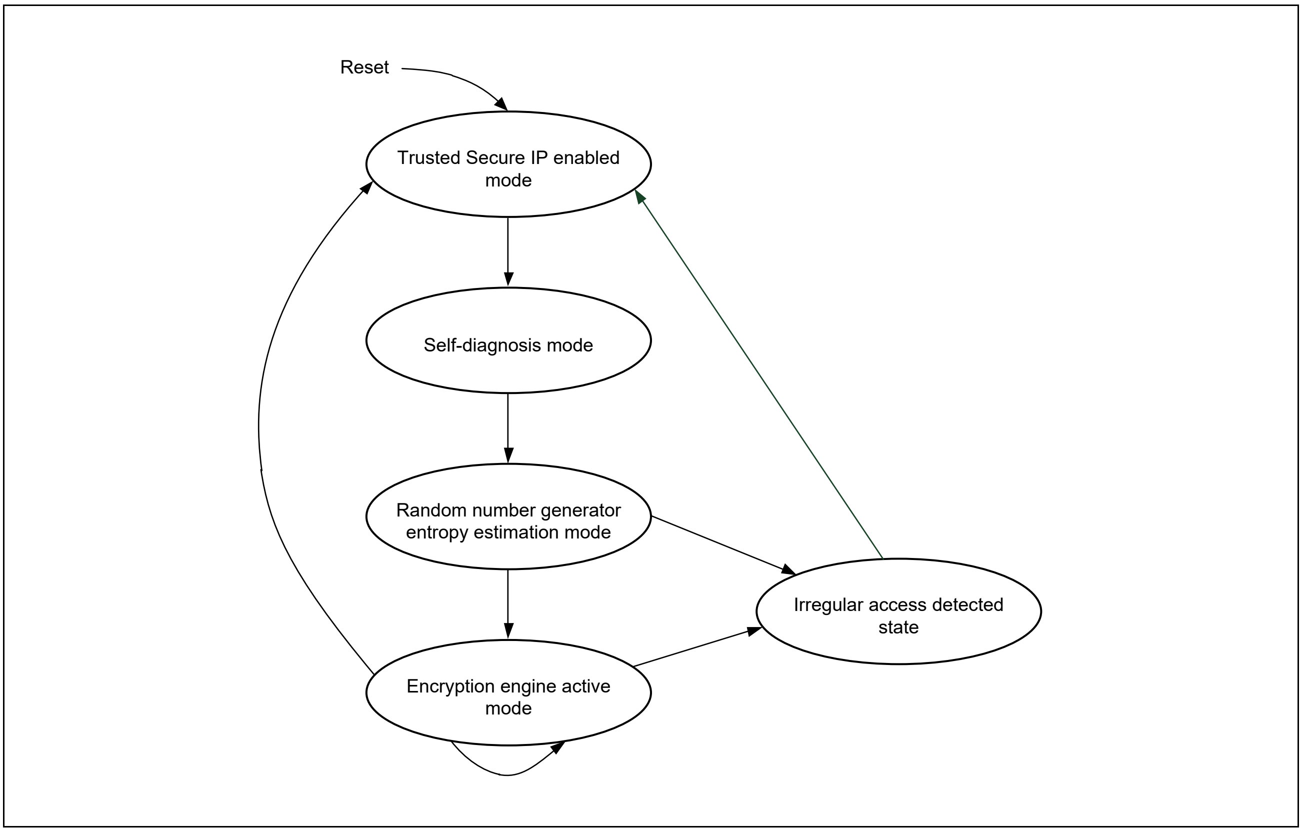


Figure 45.2 Trusted Secure IP Operating Modes and State Transitions

Many of the security functions that the Trusted Secure IP offers are applicable only in the encryption engine active mode. The operations that can be performed in this mode are given below.

(1) Key Installation

(2) Encryption and decryption

(3) Key generation

(4) Random number generation

45.2.2 Encryption Engine

Figure 45.3 shows processes of the encryption engine integrated in the Trusted Secure IP.

The encryption engine, using the key generation information, performs plaintext to ciphertext encryption and ciphertext to plaintext decryption by hardware.

In no part of the encryption or decryption process, is key data or intermediate data ever exposed outside of the Trusted Secure IP.

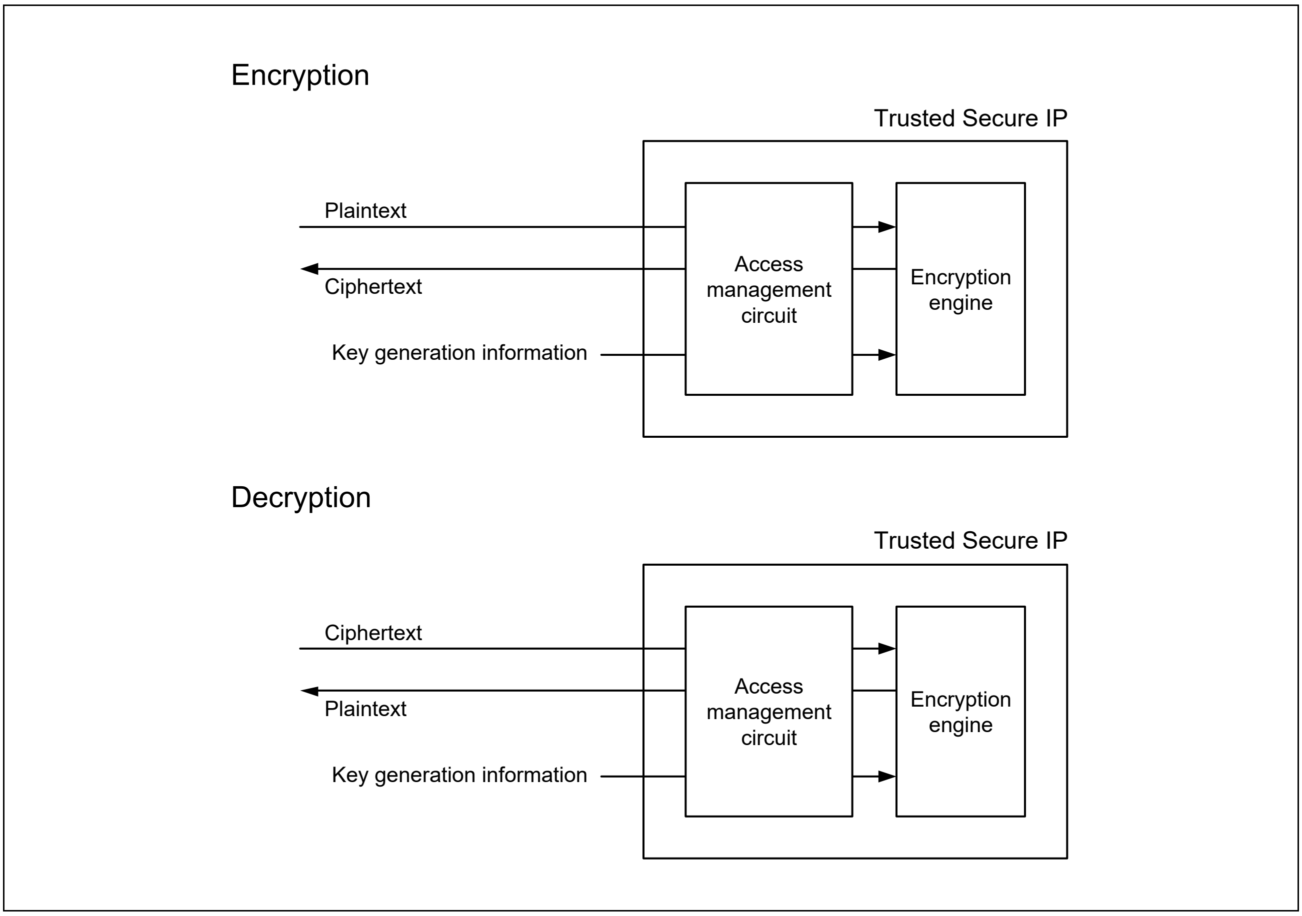


Figure 45.3 Encryption and Decryption processes by Encryption Engine

45.2.3 Key Installation

The key installation is the operation that safely converts the user key to the key generation information and stores it in flash memory. The procedure for installing the key data is given below.

(1) The user uses the key (Key-2) used for encrypting the user key to encrypt the user key (Key-1) producing eKey-1.

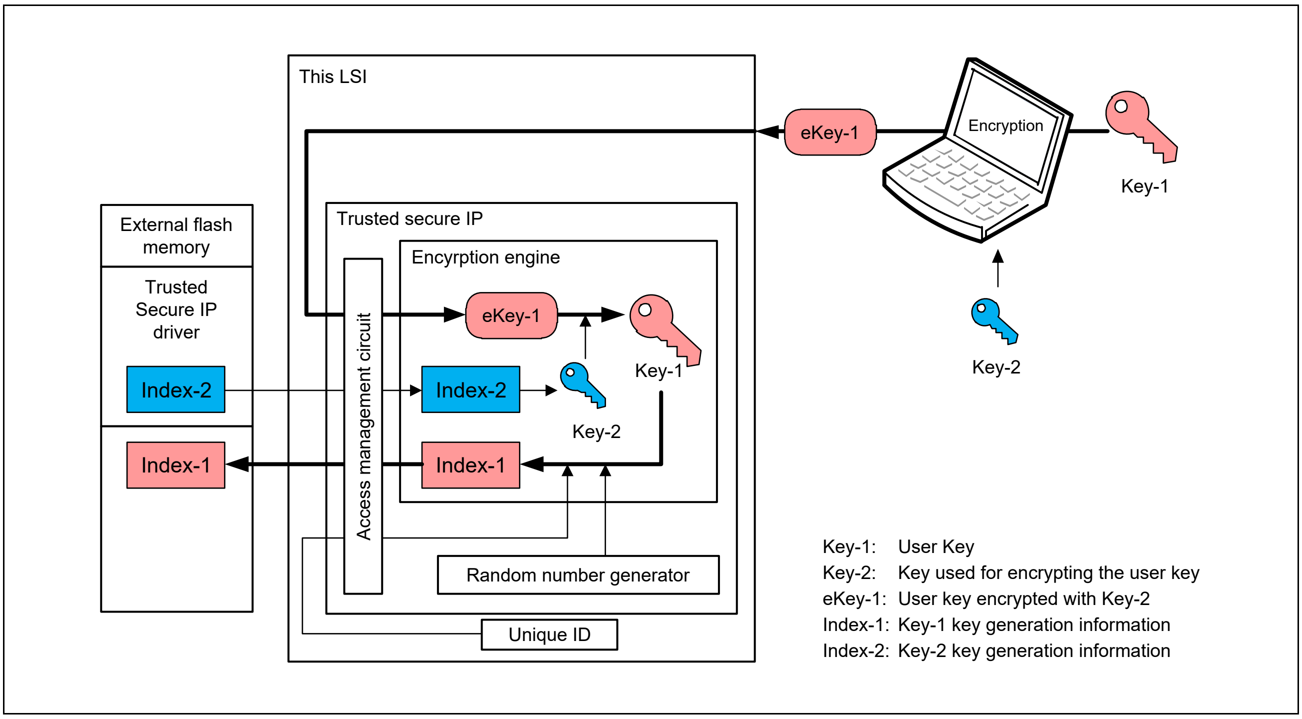
(2) The user sends the encrypted user key (eKey-1) to the Trusted Secure IP.

(3) The key generation information of the Key-2 (Index-2) contained in the Trusted Secure IP driver is used to recover the Key-2, which is then used to decrypt the user key.

(4) The user key is converted to user key generation information (Index-1) using the unique ID and a random number, and stored in flash memory.

The installation process and flow chart are given in **Figure 45.4** and **Figure 45.5**, respectively.

Once the key data is installed, the user key generation information (Index-1) can then be used to perform encryption or decryption.



Key-1: User Key
Key-2: Key used for encrypting the user key
eKey-1: User key encrypted with Key-2
Index-1: Key-1 key generation information
Index-2: Key-2 key generation information

Figure 45.4 Key Installation Process

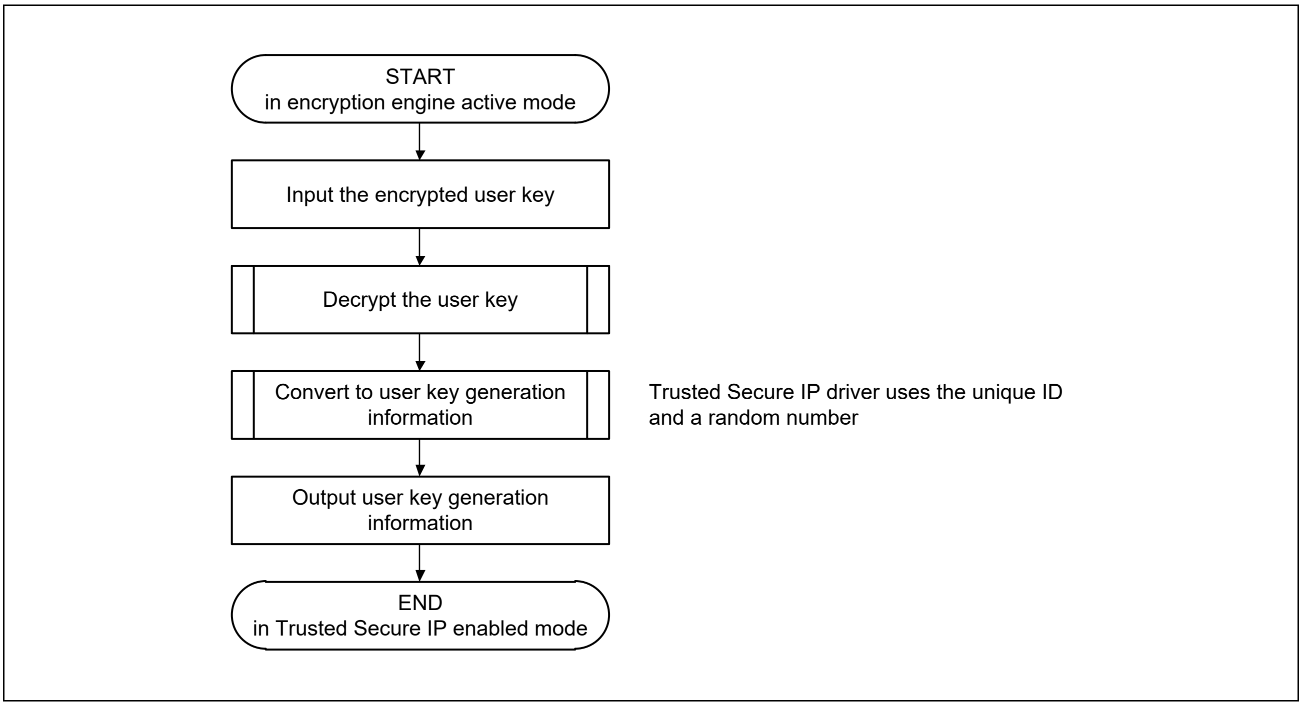


Figure 45.5 Key Installation Flow Chart

45.2.4 Encryption and Decryption

The procedures for encrypting and decrypting data are given below.

(1) Input the key generation information into the Trusted Secure IP, and recover the key data.

(2) Input the data to encrypt or decrypt into the Trusted Secure IP. This converts plaintext into ciphertext, and ciphertext into plaintext.

(3) Read the converted data.

The encryption engine has an input buffer and an output buffer, enabling encryption/decryption to proceed in parallel with data input/output.

Figure 45.6, **Figure 45.7**, and **Figure 45.8** show the timing diagram, encryption flow, and decryption flow, respectively.

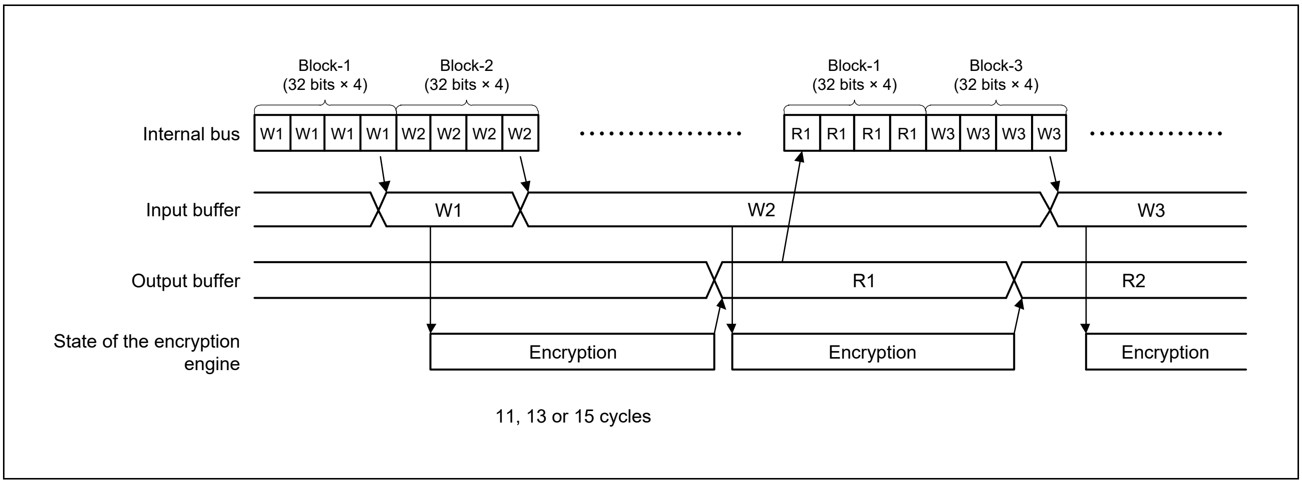


Figure 45.6 Encryption and Decryption Timing Diagram

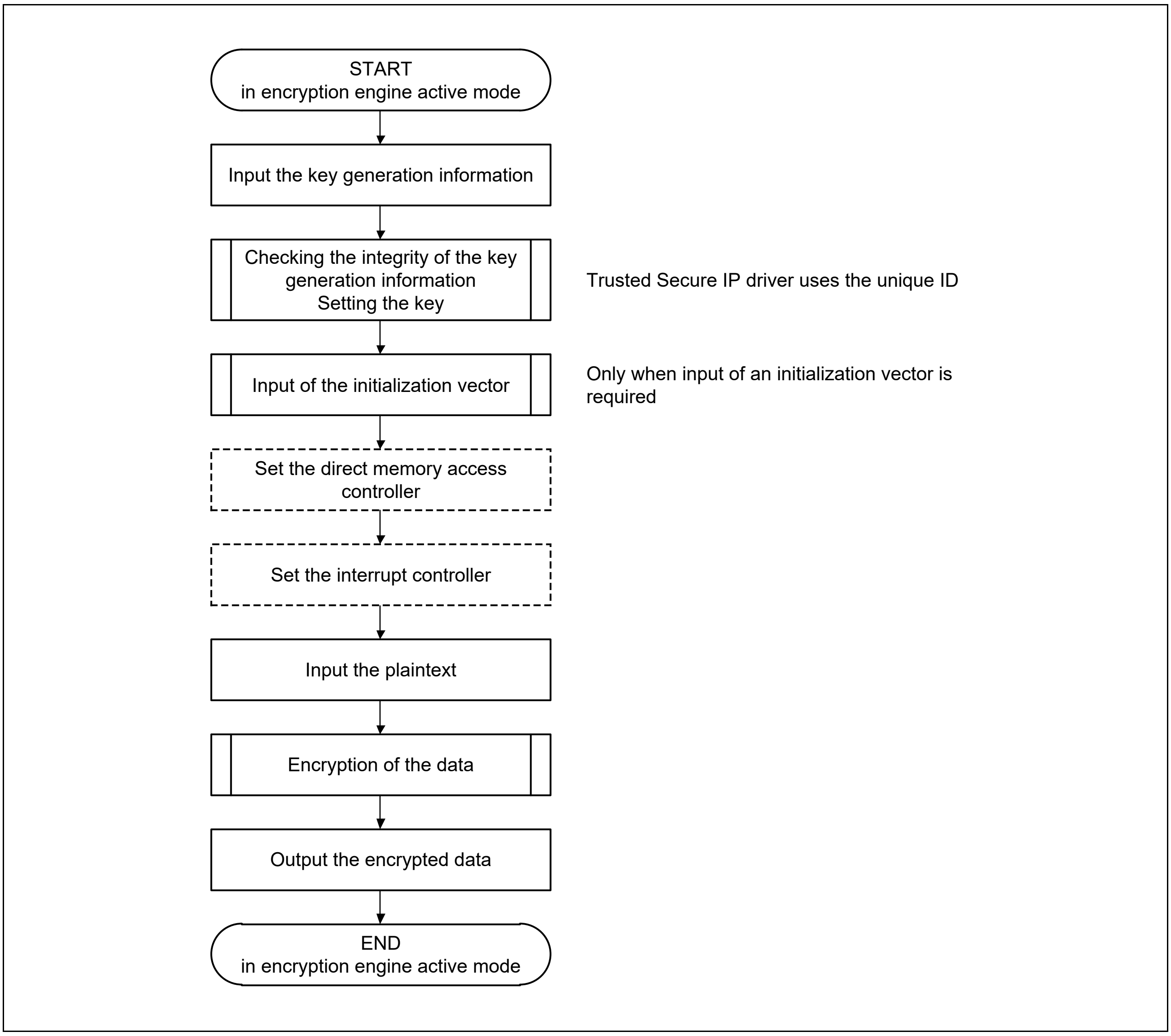


Figure 45.7 Encryption Flow Chart

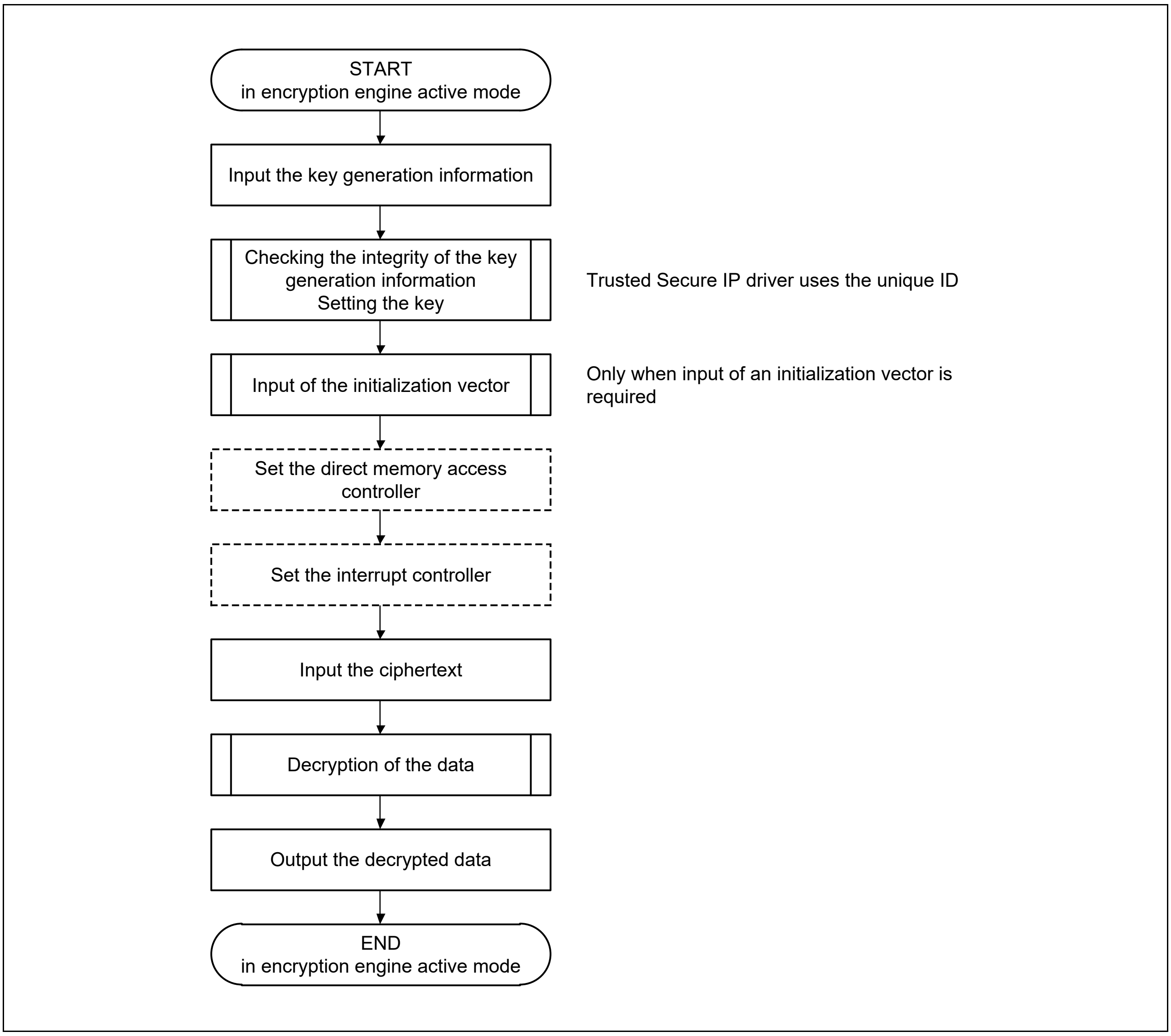


Figure 45.8 Decryption Flow Chart

45.2.5 Generating Key Generation Information (by Using Random Numbers)

Figure 45.9 shows the generating flow for the key generation information by using random numbers.

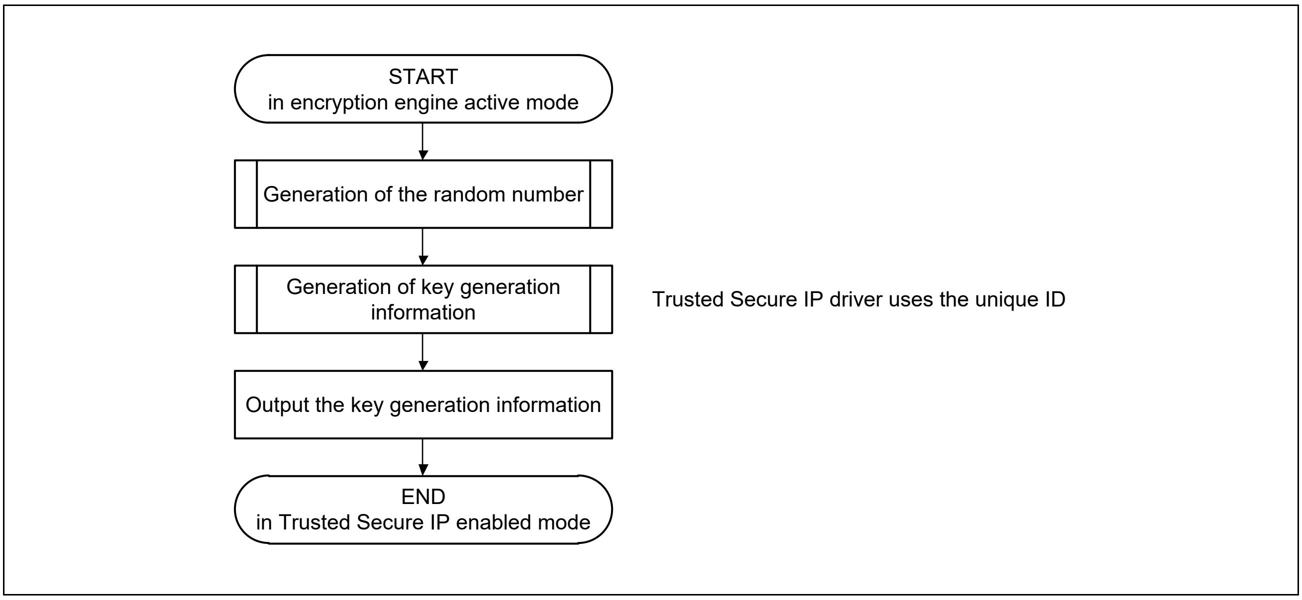


Figure 45.9 Key Generation Information Generating Flow Chart (Using Random Numbers)

45.2.6 Random Number Generation

Figure 45.10 shows the random number generation flow.

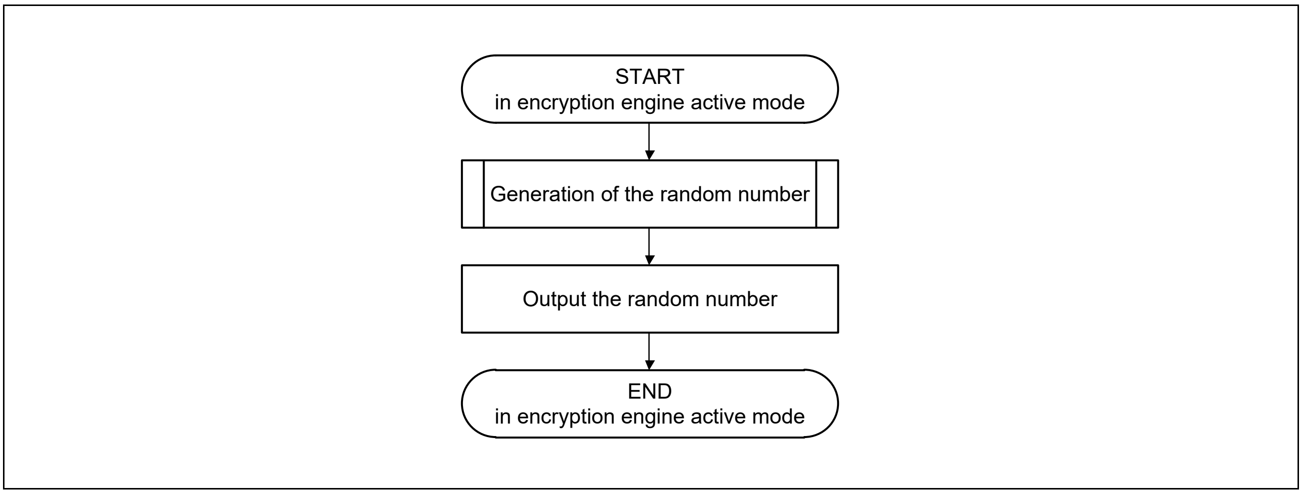


Figure 45.10 Random Number Generation Flow Chart

45.3 Interrupt

Table 45.2 lists the interrupt sources.

Trusted Secure IP driver uses interrupts caused by these interrupt sources. Do not change the setting of the interrupt controller corresponding to these interrupt sources.

Table 45.2 Trusted Secure IP Interrupt Sources

Name	Interrupt Source
PROC_BUSY	Procedure completion interrupt
ROMOK	Falsification detection interrupt
LONG_PLG	Calculation completion interrupt
WRRDY0	Write ready 0
WRRDY1	Write ready 1
WRRDY4	Write ready 4
RDRDY0	Read ready 0
RDRDY1	Read ready 1
IWRRDY	Integration write ready
IRDRDY	Integration read ready

45.4 Usage Notes

45.4.1 Trusted Secure IP Driver

Use of the Trusted Secure IP requires the Trusted Secure IP driver provided by Renesas Electronics. Please contact our sales office for information regarding the Trusted Secure IP driver.

48. Electrical Characteristics

48.1 Absolute Maximum Ratings

Table 48.1 Absolute Maximum Ratings

Item		Symbol	Value	Unit
Power supply voltage (3.3 V)		PV_{DD} USB_V_{DD33}	−0.5 to +3.8	V
Power supply voltage (1.8-V/3.3-V switchable)		SD0_PV_{DD} SD1_PV_{DD} SPI_PV_{DD}	−0.5 to +3.8	V
Power supply voltage (1.8-V/ 2.5-V/ 3.3-V switchable)		$PV_{DD182533\_0/1}$	−0.5 to +3.8	V
Power supply voltage (1.8 V)		V_{DD18} PLL1_AV_{DD18} PLL23_AV_{DD18} PLL4_AV_{DD18} PLL5_AV_{DD18} PLL6_AV_{DD18} CSI_V_{DD18} DSI_V_{DD18} USB_V_{DD18} ADC_AV_{DD18} OTP_V_{DD18}	−0.5 to +2.5	V
DDR power supply voltage (DDR4/ DDR3L switchable)		DDR_V_{DDQ}	−0.5 to +2.5	V
Power supply voltage (1.1-V)		V_{DD} PLL23_DV_{DD11} PLL5_DV_{DD11}	−0.5 to +1.5	V
Input voltage	3.3-V I/O input pins	—	−0.3 to 3.3-V power supply (PV_{DD}, USB_V_{DD33}) + 0.3	V
	1.8-V/3.3-V switchable I/O input pins	—	−0.3 to 1.8-V/3.3-V switchable power supply (SD0_PV_{DD}, SD1_PV_{DD}, SPI_PV_{DD}) + 0.3	V
	1.8-V/2.5-V/3.3-V switchable I/O input pins	—	−0.3 to 1.8-V/2.5-V/3.3-V switchable power supply ($PV_{DD182533\_0.1}$) + 0.3	V
	1.8-V I/O input pins	—	−0.3 to 1.8-V power supply (V_{DD18}, CSI_V_{DD18}, USB_V_{DD18}) + 0.3	V
Operating temperature	Ambient temperature	T_a	−40°C to +85°C*1	°C
	Junction temperature	T_j	−40°C to +125°C	°C
Storage temperature	Ambient temperature	T_{stg}	−40°C to +150°C	°C

Note 1. If wider temp is required than this range, use case has to be investigated.

48.2 Power Supply

Table 48.2 Power Supply

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Power supply voltage (3.3 V)	PV_{DD}	2.97	3.30	3.63	V	—
Power supply voltage (1.8 V)	V_{DD18}, ADC_AV_{DD18}	1.62	1.80	1.98	V	—
	OTP_V_{DD18}	1.65	1.80	1.95	V	—
Power supply voltage (1.1 V)	V_{DD}	1.05	1.10	1.15	V	—
Power supply voltage (USB)	USB_V_{DD33}	3.00	3.30	3.60	V	—
	USB_V_{DD18}	1.65	1.80	1.95	V	—
Power supply voltage (SD)	SD0_PV_{DD}, SD1_PV_{DD}	2.97	3.30	3.63	V	When 3.3 V is supplied
		1.70	1.80	1.95	V	When 1.8 V is supplied
Power supply voltage (SPI)	SPI_PV_{DD}	2.97	3.30	3.63	V	When 3.3 V is supplied
		1.70	1.80	1.95	V	When 1.8 V is supplied
Power supply voltage (Ether)	$PV_{DD182533\_0/1}$	2.97	3.30	3.63	V	When 3.3 V is supplied
		2.25	2.50	2.75	V	When 2.5 V is supplied
		1.62	1.80	1.98	V	When 1.8 V is supplied
Power supply voltage (CSI)	CSI_V_{DD18}	1.65	1.80	1.95	V	—
Power supply voltage (DSI)	DSI_V_{DD18}	1.65	1.80	1.95	V	—
Power supply voltage (DDR)	DDR_V_{DDQ}	1.14	1.20	1.26	V	When using DDR4
		1.283	1.35	1.45	V	When using DDR3L
Power supply voltage (PLL)	PLL1_AV_{DD18}, PLL23_AV_{DD18}, PLL4_AV_{DD18}, PLL5_AV_{DD18}, PLL6_AV_{DD18}	1.62	1.80	1.98	V	—
	PLL23_DV_{DD11}, PLL5_DV_{DD11}	1.05	1.10	1.15	V	—

48.3 Power-On/Power-Off Sequence

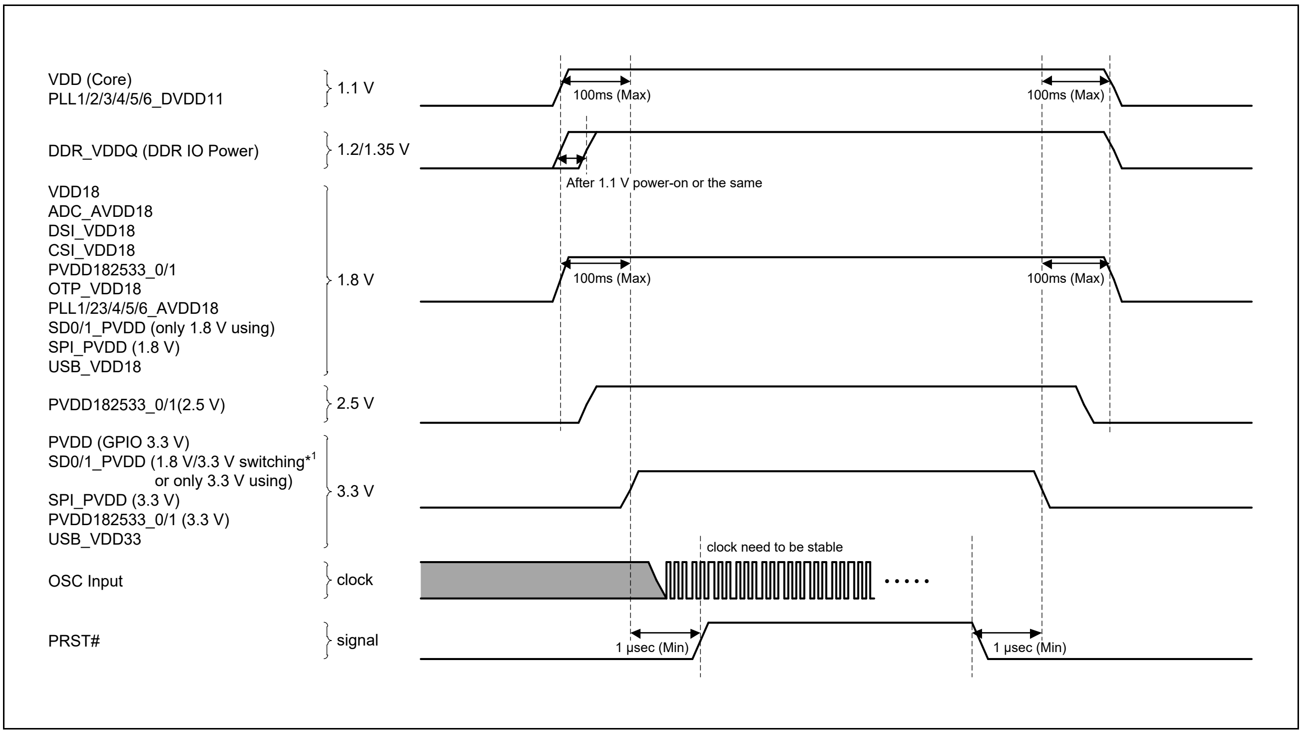


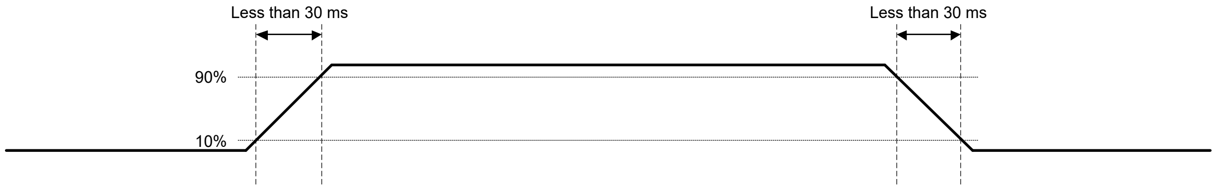
Figure 48.1 Power-On/Power-Off Sequence

NOTES

1. Turn on 3.3 V after 1.1 V/1.8 V.

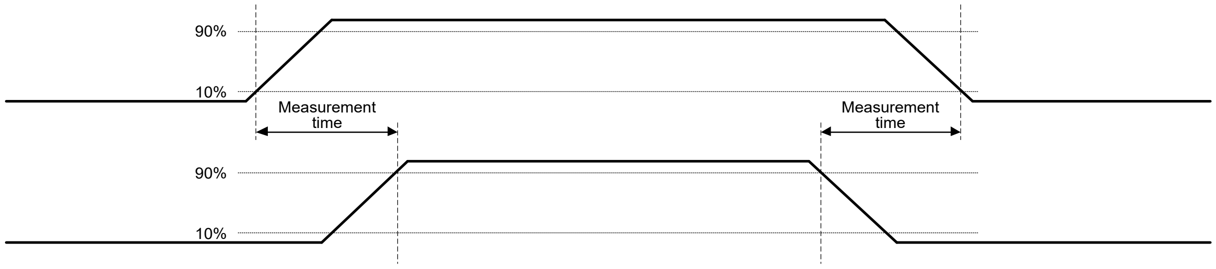
   Note 1. About SD0/1_PV$_{DD}$, especially in the case of switching between 1.8 V and 3.3 V at same power rail, 1.8-V power-on/off timing can also follow 3.3-V power rail sequence as shown in the **Figure 48.1**.

2. Turn on the DDR IO power supply at the same time as or after the 1.1-V power supply.
3. From first power rising start to last power rising end must be within 100 ms.
4. The power-off sequence is the reverse of the power-on sequence
   (PRST# → Low ⇨ 3.3 V = OFF ⇨ Other = OFF)
5. PRST# should be changed from Low to High after the 3.3-V power supply is turned on, after 1 µsec, and after the input clock from the oscillator stabilizes.
6. Refer to SD0/1_PV$_{DD}$ when connecting eMMC.



Note: The rise/fall time of each power supply should be within 30 msec.
Start point / end point of rising / falling time of each power supply should be 10% or 90% level of each power supply.

Figure 48.2 Power Up Time/Power Down Time (1)



Note: In case of there is restriction of sequence between different power supply, measurement time is starting point that is 10% of earlier rising power voltage level to end point that is 90% of later rising power voltage level, and starting point that is 90% of earlier falling power voltage level to end point that is 10% of later falling power voltage level.

Figure 48.3 Power Up Time/Power Down Time (2)

48.4 DC Characteristics

Table 48.3 DC Characteristics (1) [3.3-V I/O]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	2	—	PV_{DD} + 0.3	V	
Low-level input voltage		V_{IL}	−0.3	—	0.8	V	
Hysteresis threshold ↑		V_T+	0.9	—	2.1	V	
Hysteresis threshold ↓		V_T−	0.7	—	1.9	V	
Input hysteresis voltage		V_{HYS}	0.306	—	0.420	V	
Output logic high voltage	(I_{OH} = –2mA)	V_{OH}	PV_{DD} – 0.4	—	PV_{DD}	V	
	(I_{OH} = –4mA)						
	(I_{OH} = –8mA)						
	(I_{OH} = –12mA)						
Output logic low voltage	(I_{OL} = 2mA)	V_{OL}	0	—	0.4	V	
	(I_{OL} = 4mA)						
	(I_{OL} = 8mA)						
	(I_{OL} = 12mA)						
Weak pull-up resistor (input mode)		R_{UP}	7K	—	100K	Ω	
Weak pull-down resistor (input mode)		R_{DN}	7K	—	100K	Ω	

Note 1. Schmitt can be used at only RIIC mode.

Table 48.4 DC Characteristics (2) [1.8-V I/O]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	0.65 × V_{DD18}	—	V_{DD18} + 0.3	V	
Low-level input voltage		V_{IL}	−0.3	—	0.35 × V_{DD18}	V	
Output logic high voltage	(I_{OH} = –2mA)	V_{OH}	V_{DD18} – 0.4	—	V_{DD18}	V	
	(I_{OH}= –4mA)						
	(I_{OH} = –8mA)						
	(I_{OH} = –12mA)						
Output logic low voltage	(I_{OL} = 2mA)	V_{OL}	0	—	0.4	V	
	(I_{OL} = 4mA)						
	(I_{OL} = 8mA)						
	(I_{OL} = 12mA)						

Table 48.5 DC Characteristics (3) [3.3-V Input]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage	V_{IH}	2.3	—	PV_{DD} + 0.3	V	
Low-level input voltage	V_{IL}	−0.3	—	0.8	V	
Hysteresis threshold ↑	V_T+	0.9	—	2.1	V	
Hysteresis threshold ↓	V_T−	0.7	—	1.9	V	
Input hysteresis voltage	V_{HYS}	0.306	—	0.420	V	

Table 48.6 DC Characteristics (4) [1.8-V Input]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage	V_{IH}	0.65 × V_{DD18}	—	V_{DD18} + 0.3	V	
Low-level input voltage	V_{IL}	−0.3	—	0.35 × V_{DD18}	V	

Table 48.7 DC Characteristics (5) [RGMII/MII]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Power supply voltage	$PV_{DD182533\_n}$	2.97	3.3	3.63	V	3.3-V RGMII/MII
Input logic high	V_{IH}	2.6	—	—	V	
Input logic low	V_{IL}	—	—	0.7	V	
Output logic high voltage @ 7.7 mA	V_{OH}	2.1	—	3.6	V	
Output logic low voltage @ 9.3 mA	V_{OL}	0	—	0.5	V	
Power supply voltage	$PV_{DD182533\_n}$	2.25	2.5	2.75	V	2.5-V RGMII/MII
Input logic high	V_{IH}	1.9	—	—	V	
Input logic low	V_{IL}	—	—	0.7	V	
Output logic high voltage	V_{OH}	2.0	—	$PV_{DD182533\_n}$	V	
Output logic low voltage	V_{OL}	V_{SS}	—	0.4	V	
Power supply voltage	$PV_{DD182533\_n}$	1.62	1.8	1.98	V	1.8-V RGMII/MII
Input logic high	V_{IH}	0.7 × $PV_{DD182533\_n}$	—	$PV_{DD182533\_n}$ + 0.3	V	
Input logic low	V_{IL}	$PV_{DD182533\_n}$ − 0.3	—	0.3 × $PV_{DD182533\_n}$	V	
Output logic high @ I_{OH} = 100 µA	V_{OH}	0.85 × $PV_{DD182533\_n}$	—	$PV_{DD182533\_n}$	V	
Output logic low @ I_{OL} = 100 µA	V_{OL}	V_{SS}	—	0.15 × $PV_{DD182533\_n}$	V	

Note: n = 0, 1

Table 48.8 DC Characteristics (6) [3.3 V I/O (SD, QSPI)]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	0.625 × SDn_PV_{DD}	—	SDn_PV_{DD} + 0.3	V	
Low-level input voltage		V_{IL}	SDn_PV_{DD} − 0.3	—	0.25 × SDn_PV_{DD}	V	
Output logic high voltage	(× 0.5)	V_{OH}	0.75 × SDn_PV_{DD}	—	SDn_PV_{DD}	V	
	(× 0.75)						
	(× 1.0)						
	(× 1.5)						
Output logic low voltage	(× 0.5)	V_{OL}	0	—	0.125 × SDn_PV_{DD}	V	
	(× 0.75)						
	(× 1.0)						
	(× 1.5)						

Table 48.9 DC Characteristics (7) [1.8 V I/O (SD, QSPI)]

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
High-level input voltage		V_{IH}	1.27	—	2.00	V	
Low-level input voltage		V_{IL}	SDn_PV$_{DD}$ − 0.3	—	0.58	V	
Output logic high voltage	(× 0.5)	V_{OH}	0.7 × SDn_PV$_{DD}$	—	SDn_PV$_{DD}$	V	
	(× 0.75)						
	(× 1.0)						
	(× 1.5)						
Output logic low voltage	(× 0.5)	V_{OL}	0	—	0.3 × SDn_PV$_{DD}$	V	
	(× 0.75)						
	(× 1.0)						
	(× 1.5)						

Table 48.10 DC Characteristics(8) [I^{2}C Open Drain 3.3 V I/O]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
External pull-up supply	V_{VDDP}	2.7	—	PV$_{DD}$	V	
Input high voltage	V_{IH}	0.7 × V_{VDDP}	—	V_{VDDP} + 0.5	V	
Input Low Voltage	V_{IL}	−0.5	—	0.3 × V_{VDDP}	V	
Low level output current	I_{OL}	20	—	—	mA	V_{OL} = 0.4 V
Low level output voltage	V_{OL}	—	—	0.4	V	V_{VDDP} > 2 V
Input hysteresis	V_{HYST}	0.1 × V_{VDDP}	—	—	mV	

Table 48.11 DC Characteristics (9) [1.2-V Input (DDR4)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
DC input high level	V_{IH}(DC)	V_{REF} + 0.068	—	—	V	
DC input low level	V_{IL}(DC)	—	—	V_{REF} − 0.068	V	
AC input high level	V_{IH}(AC)	V_{REF} + 0.093	—	—	V	
AC input Low level	V_{IL}(AC)	—	—	V_{REF} − 0.093	V	
DC differential input high	V_{IHdiff}	0.136	—	—	V	
DC differential input low	V_{ILdiff}	—	—	−0.136	V	
AC differential input high	V_{IHdiff}(AC)	2 × (V_{IH}(AC) − V_{REF})	—	—	V	
AC differential input low	V_{ILdiff}(AC)	—	—	2 × (V_{REF} − V_{IL}(AC))	V	
Differential input cross point voltage relative to 0.8 × V_{DDIO} for DQS	V_{IX}(DQS)	−0.15	—	0.06	V	

Table 48.12 DC Characteristics (10) [1.35-V Input (DDR3L)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
DC input high level	V_{IH}(DC)	V_{REF} + 0.09	—	—	V	
DC input low level	V_{IL}(DC)	—	—	V_{REF} − 0.09		V_{REF} =0.5 x DDR_V_{DDQ}
AC input high level	V_{IH}(AC)	V_{REF} + 0.135	—	—	V	
AC input low level	V_{IL}(AC)	—	—	V_{REF} − 0.135	V	V_{REF} =0.5 x DDR_V_{DDQ}
DC differential input high	V_{IHdiff}	0.18	—	—	V	
DC differential input low	V_{ILdiff}	—	—	−0.18	V	
AC differential input high	V_{IHdiff}(AC)	2 × (V_{IH}(AC) − V_{REF})	—	—	V	
AC differential input low	V_{ILdiff}(AC)	—	—	2 × (V_{REF} − V_{IL}(AC))	V	V_{REF} =0.5 x DDR_V_{DDQ}
Differential input cross point voltage relative to 0.5 × DDR_VDDQ for DQS	V_{IX}(DQS)	−0.075	—	0.075	V	

Table 48.13 DC Characteristics (11) [1.2-V Output (DDR4)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
DC output high measurement level (for IV curve linearity)	V_{OH}(DC)	—	1.1 × DDR_V_{DDQ}	—	V	The swing of ± 0.15 × DDR_V_{DDQ} is based on approximately 50% of the static single-ended output peak-to-peak swing with a driver impedance of 40Ω and an effective test load of 50Ω to V_{TT} = DDR_V_{DDQ}.
DC output mid measurement level (for IV curve linearity)	V_{OM}(DC)	—	0.8 × DDR_V_{DDQ}	—	V	
DC output low measurement level (for IV curve linearity)	V_{OL}(DC)	—	0.5 × DDR_V_{DDQ}	—	V	
AC output high measurement level (for output slew rate)	V_{OH}(AC)	—	0.85 × DDR_V_{DDQ}	—	V	
AC output low measurement level (for output slew rate)	V_{OL}(AC)	—	0.55 × DDR_V_{DDQ}	—	V	

Table 48.14 DC Characteristics (12) [1.35-V Output (DDR3L)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
DC output high measurement level (for IV curve linearity)	V_{OH}(DC)	—	0.8 × DDR_V_{DDQ}	—	V	The swing of ± 0.1 × DDR_V_{DDQ} is based on approximately 50% of the static single-ended output high or low swing with a driver impedance of 40Ω and an effective test load of 25Ω to V_{TT} = DDR_V_{DDQ}/2.
DC output mid measurement level (for IV curve linearity)	V_{OM}(DC)	—	0.5 × DDR_V_{DDQ}	—	V	
DC output low measurement level (for IV curve linearity)	V_{OL}(DC)	—	0.2 × DDR_V_{DDQ}	—	V	
AC output high measurement level (for output slew rate)	V_{OH}(AC)	—	VTT + 0.1 × DDR_V_{DDQ}	—	V	
AC output low measurement level (for output slew rate)	V_{OL}(AC)	—	VTT − 0.1 × DDR_V_{DDQ}	—	V	

Table 48.15 DC Characteristics (13) [DDR4 Cross Point Voltage for Differential Output Signals (CK/DQS)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Differential output cross point voltage relative to 0.67 × DDR_VDDQ	V_{OX}(CK/DQS)	−0.06	—	0.06	V	

Table 48.16 DC Characteristics (14) [DDR3L Cross Point Voltage For Differential Output Signals (CK/DQS)]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Differential output cross point voltage relative to 0.5 × DDR_VDDQ	V_{OX}(CK/DQS)	−0.1	—	0.1	V	

Table 48.17 DC Characteristics (15) [USB 2.0]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Input levels for low/full speed						
High (driven)	V_{IH}	2.0	—	—	V	
Low	V_{IL}	—	—	0.8	V	
Differential input sensitivity	V_{DI}	0.2	—	—	V	
Differential common mode range	V_{CM}	0.8	—	2.5	V	
Input levels for high speed						
High-speed squelch detection threshold (differential signal amplitude)	V_{HSSQ}	100	—	150	mV	
High-speed data signaling common mode voltage range (guideline for receiver)	V_{HSCM}	−50	—	500	mV	
Output levels for low/full speed						
Low	V_{OL}	0.0	—	0.3	V	
High (driven)	V_{OH}	2.8	—	3.6	V	
Output signal crossover voltage	V_{CRS}	1.3	—	2.0	V	
Output levels for high-speed						
High-speed idle level	V_{HSOI}	−10.0	—	10.0	mV	
High-speed data signaling high	V_{HSOH}	360	—	440	mV	
High-speed data signaling low	V_{HSOL}	−10.0	—	10.0	mV	
Chirp J level (differential voltage)	V_{CHIRPJ}	700	—	1100	mV	
Chirp K level (differential voltage)	V_{CHIRPK}	−900	—	−500	mV	

Table 48.18 DC Characteristics (16) [ADC]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Resolution	—	—	12	—	Bit	
Analog input channel	—	—	—	8	Channel	
Analog input range	AIN	V_{SS}	—	ADC_AV_{DD18}	V	
Differential non-linearity	DNL	—	—	±3.0	LSB	
Integral non-linearity	INL	—	—	±6.0	LSB	

Table 48.19 DC Characteristics (17) [Current Consumption]

Item	Power rail symbol	Max Current	Unit	Remarks
Power Supply voltage(3.3V)	PV_{DD}	620	mA	PV_{DD}=3.6V
Power Supply voltage(1.8V)	V_{DD18}	30	mA	V_{DD18}=1.98V
	OTP_AV_{DD18}	20	mA	OTP_V_{DD18}=1.95V
	ADC_AV_{DD18}	10	mA	ADC_AV_{DD18}=1.98V
Power Supply voltage(1.1V)	V_{DD}	4060	mA	V_{DD}=1.15V, Condition: Cortex-A55, Dhrystone
Power supply voltage (USB/3.3V)	USB_V_{DD33}	20	mA	USB_V_{DD33}=3.6V
Power supply voltage (USB/1.8V)	USB_V_{DD18}	140	mA	USB_V_{DD18}=1.95V
Power supply voltage (SD/3.3V)	SD0_PV_{DD}	340	mA	SD0_PV_{DD}=3.63V
	SD1_PV_{DD}	220	mA	SD1_PV_{DD}=3.63V
Power supply voltage (SD/1.8V)	SD0_PV_{DD}	110	mA	SD0_PV_{DD}=1.95V
	SD1_PV_{DD}	70	mA	SD1_PV_{DD}=1.95V
Power supply voltage (SPI/3.3V)	SPI_PV_{DD}	310	mA	SPI_PV_{DD}=3.63V
Power supply voltage (SPI/1.8V)	SPI_PV_{DD}	100	mA	SPI_PV_{DD}=1.95V
Power supply voltage (Ether/3.3V)	$PV_{DD182533\_n}$	380	mA	$PV_{DD182533\_n}$=3.63V when 2 channels are in use
Power supply voltage (Ether/2.5V)	$PV_{DD182533\_n}$	240	mA	$PV_{DD182533\_n}$=2.75V when 2 channels are in use
Power supply voltage (Ether/1.8V)	$PV_{DD182533\_n}$	140	mA	$PV_{DD182533\_n}$=1.98V when 2 channels are in use
Power supply voltage (CSI/1.8V)	CSI_V_{DD18}	100	mA	CSI_V_{DD18}=1.95V
Power supply voltage (DSI/1.8V)	DSI_V_{DD18}	100	mA	DSI_V_{DD18}=1.95V
Power supply voltage (DDR/1.2V)	DDR_V_{DDQ}	220	mA	DDR_V_{DDQ}=1.26V
Power supply voltage (DDR/1.35V)	DDR_V_{DDQ}	210	mA	DDR_V_{DDQ}=1.45V
Power supply voltage (PLL/1.8V)	PLL1_AV_{DD18}	10	mA	PLL1_AV_{DD18}=1.98V
	PLL23_AV_{DD18}	20	mA	PLL23_AV_{DD18}=1.98V
	PLL4_AV_{DD18}	10	mA	PLL4_AV_{DD18}=1.98V
	PLL5_AV_{DD18}	10	mA	PLL5_AV_{DD18}=1.98V
	PLL6_AV_{DD18}	10	mA	PLL6_AV_{DD18}=1.98V
Power supply voltage (PLL/1.1V)	PLL23_DV_{DD11}	20	mA	PLL23_DV_{DD11}=1.15V
	PLL5_DV_{DD11}	10	mA	PLL5_DV_{DD11}=1.15V

Note: T_j = 125°C, n = 0, 1

48.5 AC Characteristics

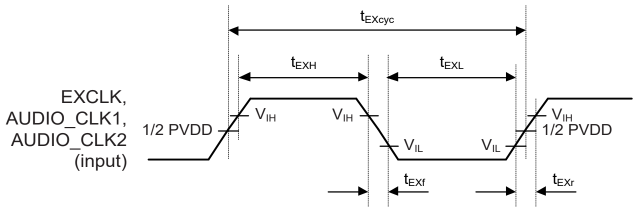
Conditions: V_{DD} = PLL23_DV_{DD11} = PLL5_DV_{DD11} = 1.1 ± 0.05 V,
PLL1_AV_{DD18} = PLL23_AV_{DD18} = PLL4_AV_{DD18} = PLL6_AV_{DD18} = 1.8 ± 0.18 V,
ADC_AV_{DD18} = V_{DD18} = 1.8 ± 0.18 V, OTP_V_{DD18} = 1.8 V ± 0.15 V,
USB_V_{DD18} = CSI_V_{DD18} = DSI_V_{DD18} = 1.8 ± 0.15 V,
PV_{DD} = 3.3 ± 0.33 V, USB_V_{DD33} = 3.3 ± 0.3 V, DDR_V_{DDQ} = 1.2 ± 0.06 V/1.283 to 1.45 V,
SPI_PV_{DD} = SDn_PV_{DD} (n = 0, 1) = 3.3 ± 0.33 V/1.70 to 1.95 V,
$PV_{DD182533\_n}$ (n = 0, 1) = 3.3 ± 0.33 V/2.5 ± 0.25 V/1.8 ± 0.18 V, V_{SS} = 0 V,
T_a = −40 to +85°C, T_j = −40 to +125°C

48.5.1 Clock Timing

Table 48.20 Clock Timing Table

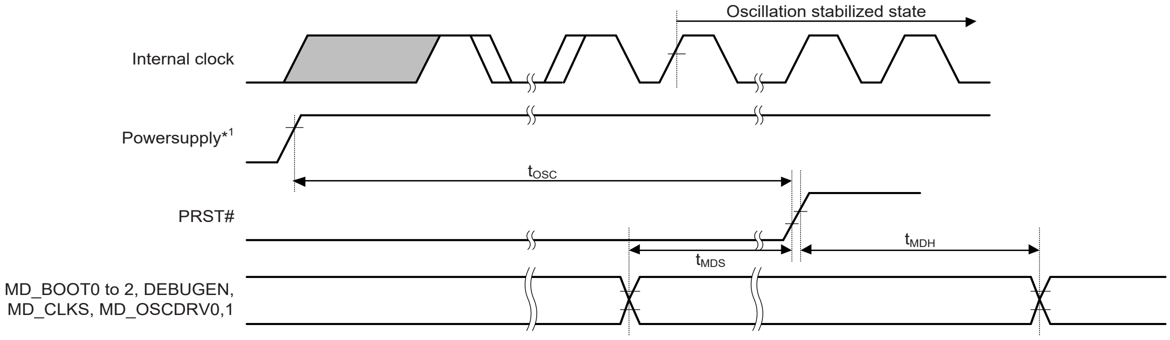
Item	Symbol	Min.	Max.	Unit	Figures
EXCLK clock input frequency	f_{EX}	24 – 50ppm*[1]	24 + 50ppm*[1]	MHz	**Figure 48.4**
EXCLK clock input cycle time	t_{EXcyc}	41.67	41.67	ns	
AUDIO_CLK1, AUDIO_CLK2 clock input frequency (external clock is input)	f_{EX}	10	50	MHz	
AUDIO_CLK1, AUDIO_CLK2 clock input cycle time (external clock is input)	t_{EXcyc}	20	100	ns	
EXCLK, AUDIO_CLK1, AUDIO_CLK2 clock input low level pulse width	t_{EXL}	0.4	0.6	t_{EXcyc}	
EXCLK, AUDIO_CLK1, AUDIO_CLK2 clock input high level pulse width	t_{EXH}	0.4	0.6	t_{EXcyc}	
EXCLK, AUDIO_CLK1, AUDIO_CLK2 clock input rise time	t_{EXr}	—	4	ns	
EXCLK, AUDIO_CLK1, AUDIO_CLK2 clock input fall time	t_{EXf}	—	4	ns	
Oscillator stabilization time	t_{OSC}	—	1	ms	**Figure 48.5**, **Figure 48.6**
Mode hold time	t_{MDH}	—	100	ns	
Mode setup time	t_{MDS}	—	100	ns	

Note 1. When using RGMII interface. If not using RGMII mode, this spec is ±100 ppm.



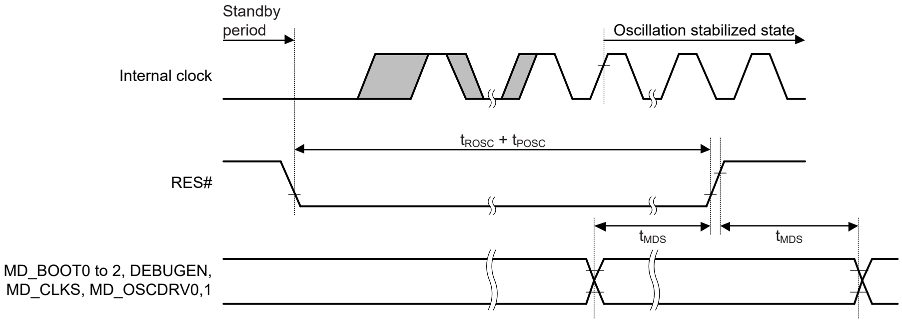
Note: When the clock is input on the EXCLK, AUDIO_CLK1 or AUDIO_CLK2

Figure 48.4 EXCLK, AUDIO_CLK1 and AUDIO_CLK2 Clock Input Timing



Note 1. V_{DD}, PLLn_DV_{DD11}, PLLn_AV_{DD18}, ADC_AV_{DD18}, V_{DD18}, OTP_V_{DD18}, USB_V_{DD18}, CSI_V_{DD18}, DSI_V_{DD18}, PV_{DD},USB_V_{DD33}, DDR_V_{DDQ}, SPI_PV_{DD}, SDn_PV_{DD}, $PV_{DD182533}$_0, $PV_{DD182533}$_1, OTP_V_{DD18}

Figure 48.5 Power-On Oscillation Settling Time



Note: Oscillation setting time when the internal oscillator is used.

Figure 48.6 Oscillation Settling Time on Return from Standby (Return by Reset)

48.5.2 SDHI Access Timing

48.5.2.1 SDHI Access Timing (SDR 3.3-V)

Table 48.21 SDHC AC Access Timing (SDR at 3.3-V Operation)

		Default Speed Mode (16.67 MHz)		High Speed Mode (33.33 MHz)			
Item	Symbol	Min.	Max.	Min.	Max.	Unit	Figures
SD_CLK clock cycle	t_{SDCYC}	60.00	—	30.0	—	ns	**Figure 48.7**
SD_CLK clock high level width	t_{SDWH}	10	—	7	—	ns	
SD_CLK clock low level width	t_{SDWL}	10	—	7	—	ns	
SD_CLK clock rise time	t_{SDLH}	—	10	—	3	ns	
SD_CLK clock fall time	t_{SDHL}	—	10	—	3	ns	
SD_CMD,SD_DATA output delay	t_{SDODLY}	−9.0	4.0	−8.0	4.0	ns	
SD_CMD,SD_DATA input set up time	t_{SDIS}	5.5	—	5.5	—	ns	
SD_CMD,SD_DATA input hold time	t_{SDIH}	2.0	—	2.0	—	ns	
SD_CMD,SD_DATA input data width	t_{SDIDW}	—	—	—	—	ns	

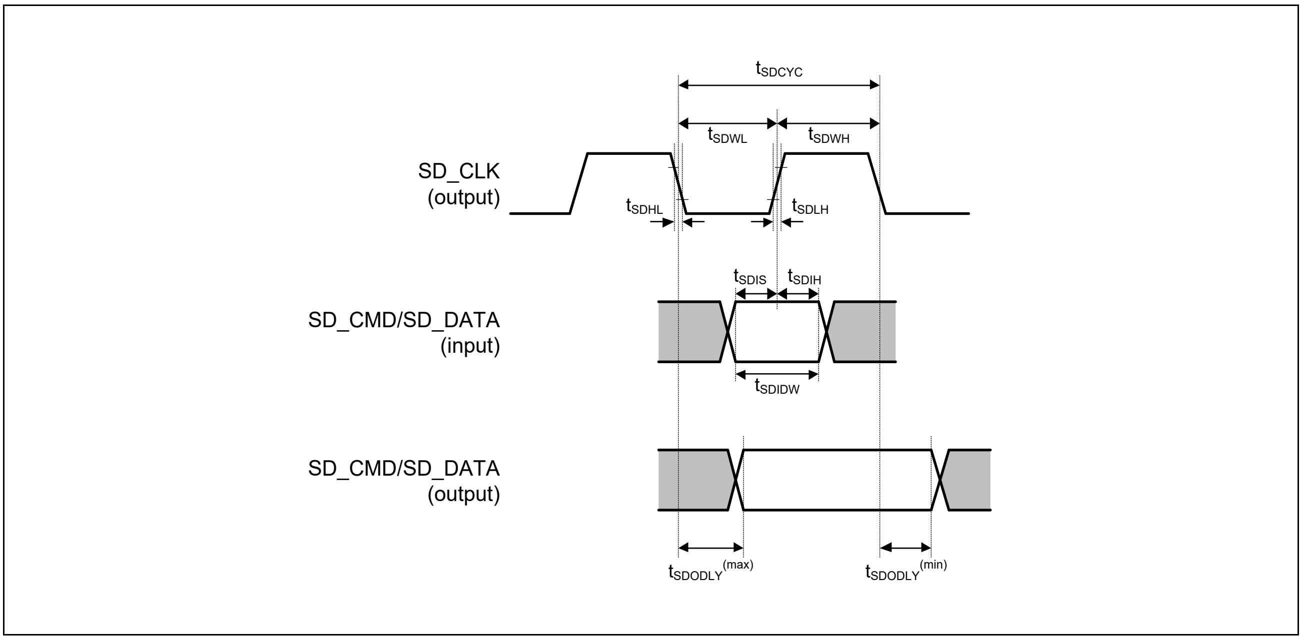


Figure 48.7 SDHC Interface Timing (SDR 3.3-V Power Supply)

NOTE

The disclosure of other characteristics of the SD interface needs the conclusion of the following agreement.

- SD Host/Ancillary Product License Agreement (SD HALA)

For details, contact your local sales representatives.

48.5.3 eMMC Access Timing

48.5.3.1 eMMC Host Interface Timing (Default)

Table 48.22 eMMC Host Interface Timing (MMC Default 3.3-V Power Supply)

Item	Symbol	Min.	Max.	Unit	Figures
SD0_CLK clock cycle	t_{MMCPP}	30.00	—	ns	**Figure 48.8**
SD0_CLK clock high level width	t_{MMCWH}	7	—	ns	
SD0_CLK clock low level width	t_{MMCWL}	7	—	ns	
SD0_CLK clock rise time	t_{MMCLH}	—	3	ns	
SD0_CLK clock fall time	t_{MMCHL}	—	3	ns	
SD0_CMD/SDDAT output delay	$t_{MMCODLY}$	−8.0	4.0	ns	
SD0_CMD/SDDAT input set up time	t_{MMCISU}	5.5	—	ns	
SD0_CMD/SDDAT input hold time	t_{MMCIH}	2.0	—	ns	
SD0_CMD/SDDAT input data width	t_{MMCIDW}	—	—	ns	

Table 48.23 eMMC Host Interface Timing (MMC Default 1.8-V Power Supply)

Item	Symbol	Min.	Max.	Unit	Figures
SD0_CLK clock cycle	t_{MMCPP}	15.00	—	ns	**Figure 48.8**
SD0_CLK clock high level width	t_{MMCWH}	4.5	—	ns	
SD0_CLK clock low level width	t_{MMCWL}	4.5	—	ns	
SD0_CLK clock rise time	t_{MMCLH}	—	2.45	ns	
SD0_CLK clock fall time	t_{MMCHL}	—	2.45	ns	
SD0_CMD/SDDAT output delay	$t_{MMCODLY}$	−6.5	3.50	ns	
SD0_CMD/SDDAT input set up time	t_{MMCISU}	4.00	—	ns	
SD0_CMD/SDDAT input hold time	t_{MMCIH}	1.40	—	ns	
SD0_CMD/SDDAT input data width	t_{MMCIDW}	—	—	ns	

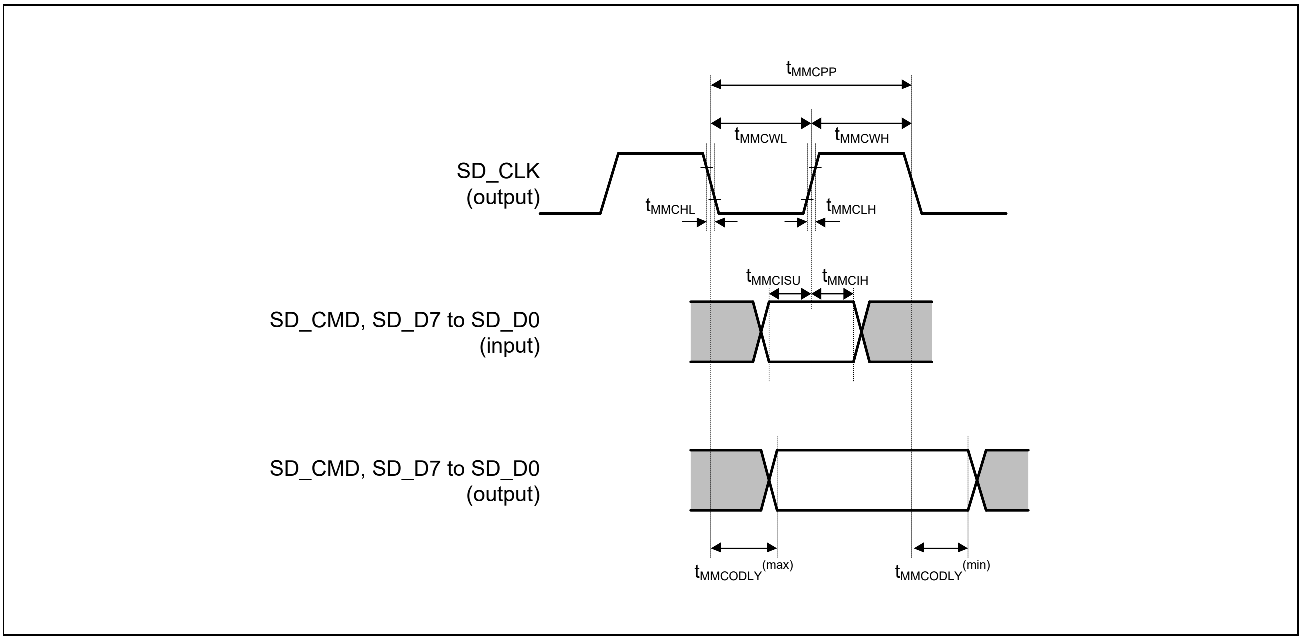


Figure 48.8 eMMC Host Interface Timing (MMC Default 1.8-V/3.3-V Power Supply)

48.5.3.2 eMMC host interface timing (HS-SDR)

NOTES

1. The spec of eMMC host interface timing (HS-SDR 3.3-V power supply) is the same as **Table 48.22** eMMC host interface timing (MMC default 3.3-V power supply).
2. The spec of eMMC host interface timing (HS-SDR 1.8V power supply) is the same as **Table 48.23** eMMC host interface timing (MMC default 1.8-V power supply)

48.5.3.3 eMMC host interface timing (HS200)

Table 48.24 eMMC Host Interface Timing (HS200 1.8-V Power Supply Operation, Output Load 15 pF)

Item	Symbol	Min.	Max.	Unit	Figures
SD0_CLK clock cycle	t_{MMCPP}	7.50	15.0	ns	**Figure 48.9**
SD0_CLK clock high level width	t_{MMCWH}	2.25	—	ns	
SD0_CLK clock low level width	t_{MMCWL}	2.25	—	ns	
SD0_CLK clock rise time	t_{MMCLH}	—	1.22	ns	
SD0_CLK clock fall time	t_{MMCHL}	—	1.22	ns	
SD0_CMD/SDDAT output delay	$t_{MMCODLY}$	−2.8	1.80	ns	
SD0_CMD/SDDAT input set up time	t_{MMCISU}	—	—	ns	
SD0_CMD/SDDAT input hold time	t_{MMCIH}	—	—	ns	
SD0_CMD/SDDAT input data width	t_{MMCIDW}	4.31	—	ns	

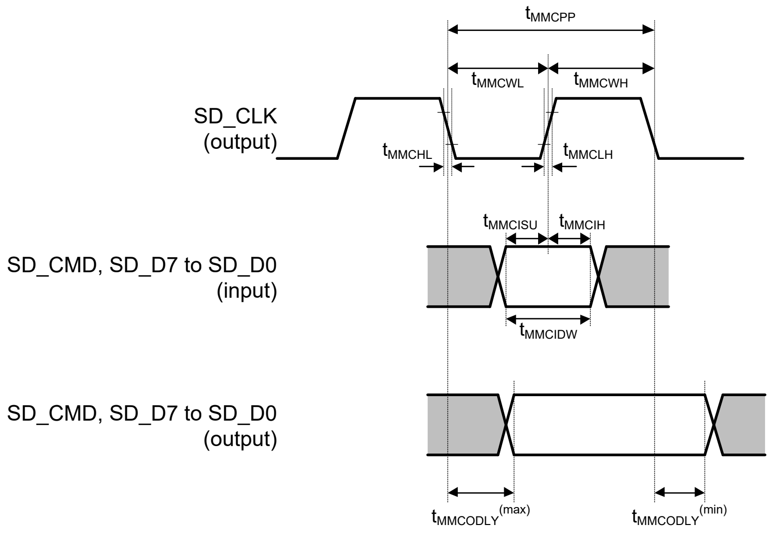


Figure 48.9 eMMC Host Interface (MMC Interface HS200 Mode 1.8-V Power Supply Selection)

48.5.4 CRU Access Timing

Table 48.25 CRU AC Access Timing

Item	Symbol	Min.	Max.	Unit	Figures
CAM_PCLK clock input cycle time	t_{CCYC}	13.4	74	ns	**Figure 48.10**
CAM_PCLK clock input rise time	t_{CR}	—	1	ns	
CAM_PCLK clock input low level pulse width	t_{CLW}	0.4	0.6	t_{CCYC}	
CAM_PCLK clock input high level pulse width	t_{CHW}	0.4	0.6	t_{CCYC}	
CAM_DATA0-15, CAM_HREF, CAM_VSYNC, CAM_FIELD setup time (P42_4 input)	t_{CSU1}	2	—	ns	
CAM_FIELD setup time (P3_1 input)	t_{CSU2}	4	—	ns	
CAM_DATA0-15, CAM_HREF, CAM_VSYNC, CAM_FIELD hold time	t_{CHD}	2	—	ns	

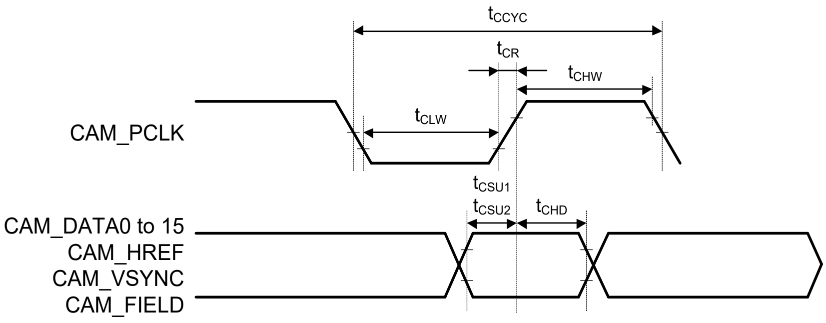


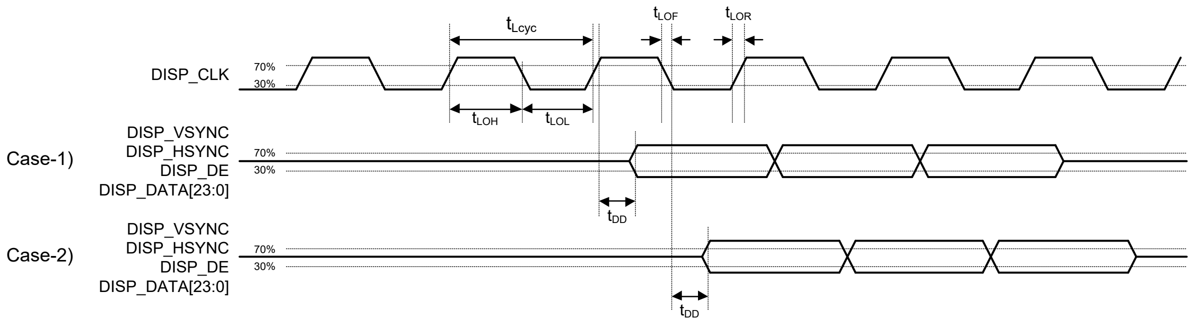
Figure 48.10 CRU AC Access Timing

48.5.5 LCDC Access Timing

Table 48.26 LCDC AC Access Timing

Item	Symbol	Min.	Max.	Unit	Figures
DCLK period	t_{Lcyc}	11.5	172.33	ns	**Figure 48.11**
DCLK low pulse width	t_{LOL}	$t_{Lcyc}/2 - 1.06$	$t_{Lcyc}/2 + 1.06$	ns	
DCLK high pulse width	t_{LOH}	$t_{Lcyc}/2 - 1.06$	$t_{Lcyc}/2 + 1.06$	ns	
DCLK rise time	t_{LOR}	—	3	ns	
DCLK fall time	t_{LOF}	—	3	ns	
Data output dealy	t_{DD}	−1.5	1.5	ns	

Note: AC access timing condition: SR = 1 (fast) / Drive capacity setting 12 mA / Load capacity 30 pF.



Note: *Case-1*) DISP_CLK forward rotation output mode
When DPI_CLKMD bit in DU_DITR0 register is 0.

Case-2) DISP_CLK inverted output mode
When DPI_CLKMD bit in DU_DITR0 register is 1.

Figure 48.11 LCDC AC Access Timing

48.5.6 USB 2.0 Host/Function Module Access Timing

48.5.6.1 USB 2.0 Low-Speed Access Timing

Table 48.27 USB Transceiver Timing (Low-Speed)

Item	Symbol	Min.	Max.	Unit	Figures
Rise time	t_{LR}	75	300	ns	**Figure 48.12**
Fall time	t_{LF}	75	300	ns	
Rise/fall time lag	t_{LR}/t_{LF}	80	125	%	

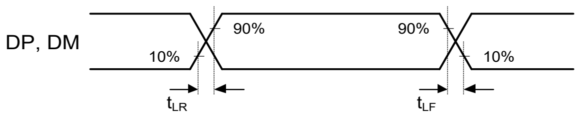
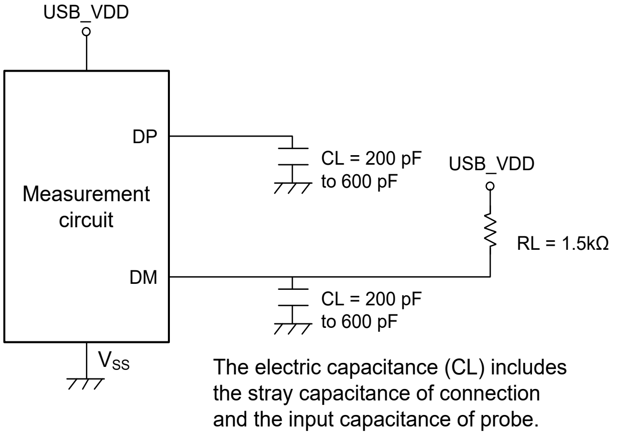


Figure 48.12 USB0_DP, USB1_DP, USB0_DM, and USB1_DM Output Timing (Low-Speed)



The electric capacitance (CL) includes the stray capacitance of connection and the input capacitance of probe.

Figure 48.13 Measurement Circuit (Low-Speed)

48.5.6.2 USB 2.0Full-Speed Access Timing

Table 48.28 USB Transceiver Timing (Full-Speed)

Item	Symbol	Min.	Max.	Unit	Figures
Rise time	t_{FR}	4	20	ns	**Figure 48.14**
Fall time	t_{FF}	4	20	ns	
Rise/fall time lag	t_{FR}/t_{FF}	90	111.11	%	

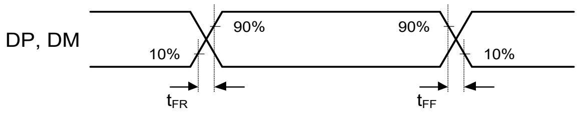
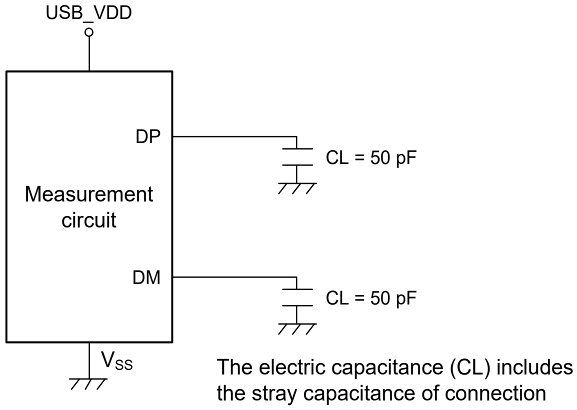


Figure 48.14 USB0_DP, USB1_DP, USB0_DM, and USB1_DM Output Timing (Full-Speed)



The electric capacitance (CL) includes the stray capacitance of connection and the input capacitance of probe.

Figure 48.15 Measurement Circuit (Full-Speed)

48.5.6.3 USB 2.0 Hi-Speed Access Timing

Table 48.29 USB Transceiver Timing (Hi-Speed)

Item	Symbol	Min.	Max.	Unit	Figures
Rise edge rate	t_{HSR}	—	2133	V/μs	**Figure 48.16**
Fall edge rate	t_{HSF}	—	2133	V/μs	
Output driver resistance	Z_{HSDRV}	40.5	49.5	Ω	

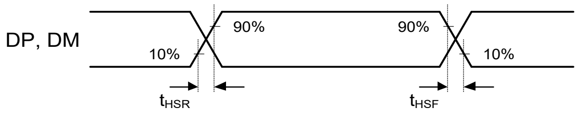


Figure 48.16 USB0_DP, USB1_DP, USB0_DM, and USB1_DM Output Timing (Hi-Speed)

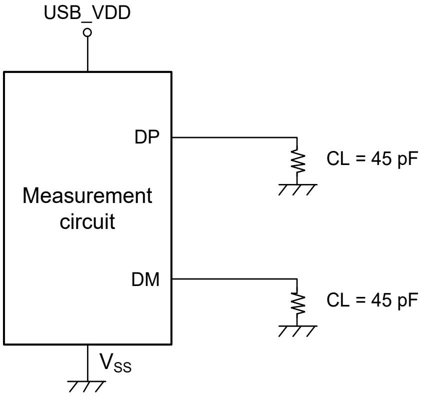
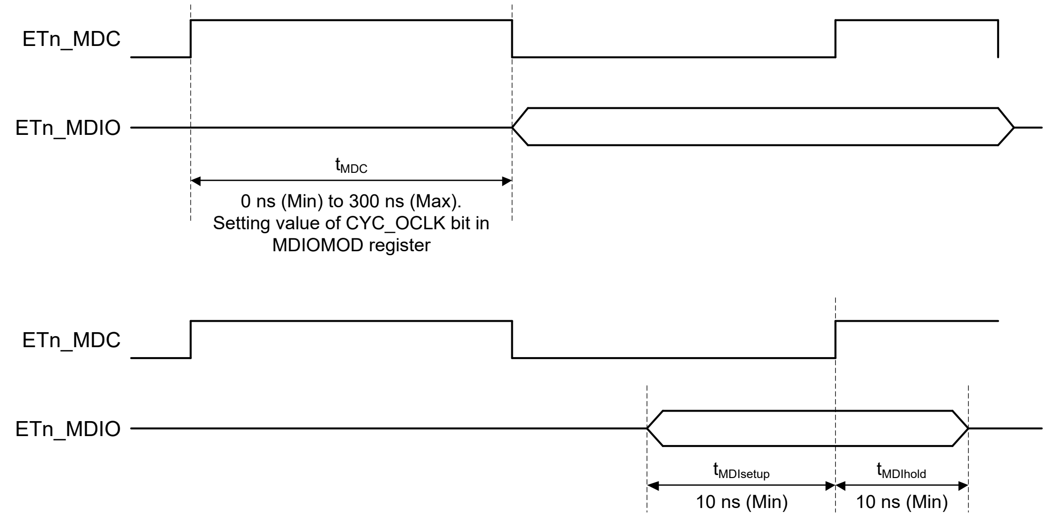


Figure 48.17 Measurement Circuit (Hi-Speed)

48.5.7 Ethernet Interface Access Timing

Table 48.30 Ethernet Interface Access Timing

Item	Symbol	Min.	Max.	Unit	Figures
MDC half cycle	t_{MDC}	0	300	ns	**Figure 48.18**
MDI setup time	$t_{MDIsetup}$	10	—	ns	
MDI hold time	$t_{MDIhold}$	10	—	ns	



Note: (n = 0, 1)

Figure 48.18 Management Interface

48.5.7.1 Ethernet-IF (Ether MII)

Table 48.31 Ethernet-IF Access Timing (Ether MII)

Item		Symbol	Min.	Max.	Unit	Figures
Ether MII	ETH_GTXTXC_TXC period	t_{Tcyc}	40	—	ns	**Figure 48.19**
	ETH_TXCTL output delay	t_{TENd}	0	25	ns	
	ETH_TXD3-0 output delay	t_{MTDd}	0	25	ns	
	ETH_RXC period	t_{Rcyc}	40	—	ns	
	ETH_RXDV setup time	t_{RDVs}	10	—	ns	
	ETH_RXDV hold time	t_{RDVh}	10	—	ns	
	ETH_RXD3-0 setup time	t_{MRDs}	10	—	ns	
	ETH_RXD3-0 hold time	t_{MRDh}	10	—	ns	
	ETH_RXER setup time	t_{RERs}	10	—	ns	
	ETH_RXER hold time	t_{RERh}	10	—	ns	

Note: I/O driving ability: 8 mA
CL = 8 pF

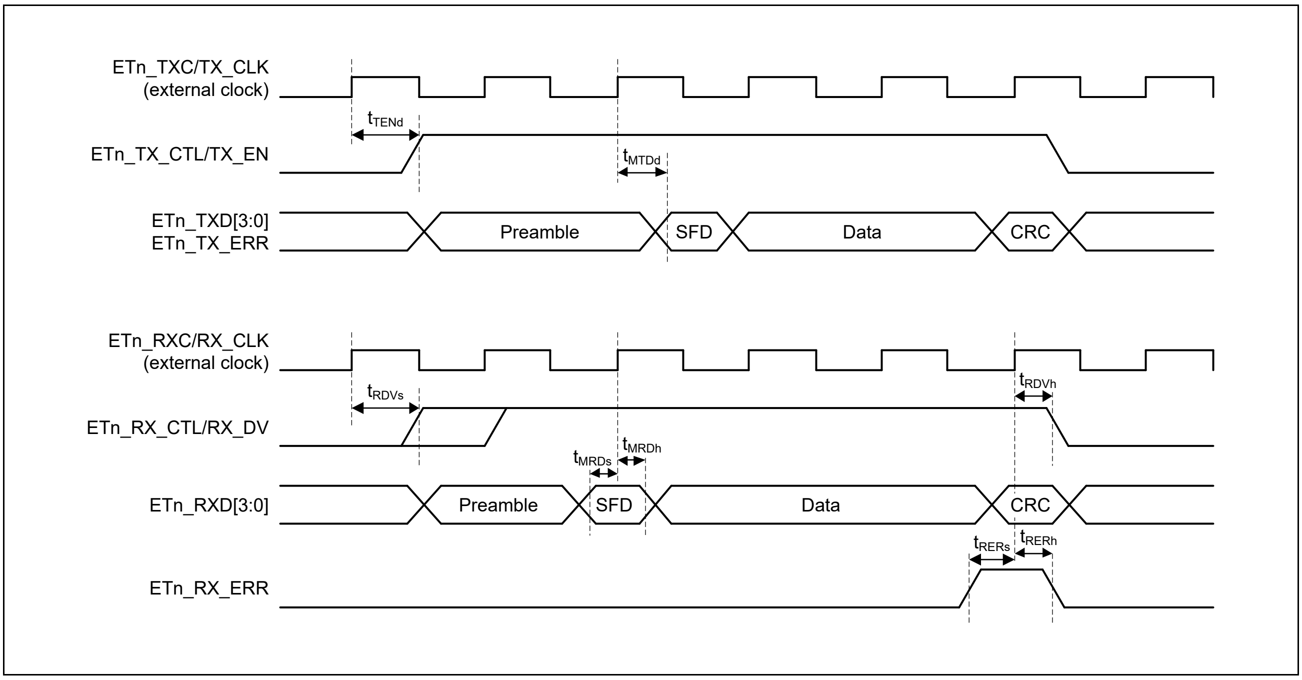


Figure 48.19 MII Transmission Timing

Validated with IEEE802.3 regulation.

The propagation delay for each twisted pair, measured from the MII connector to the PHY, shall not exceed 2.5 ns.

The variation in the propagation delay of the twisted pairs in a given cable bundle, measured from the MII connector to the PHY, shall not exceed 0.1 ns.

48.5.7.2 Ethernet-IF (Ether RGMII)

Table 48.32 Ethernet-IF Access Timing (Ether RGMII)

Item		Symbol	Min.	Typ.	Max.	Unit	capaci-tance	Remarks	Figures
Ether RGMII	Data to clock output skew @ transmitter	T_{skewT}	−500	0	500	ps	8 pF	Tx RGMII	**Figure 48.20**
	Data to clock input skew @ receiver	T_{skewR}	1	1.8	2.6	ns	8 pF		
	Data to clock output setup @ transmitter integrated delay	T_{setupT}	1.2	2.0	—	ns	8 pF	Rx RGMII-ID	**Figure 48.21**
	Clock to data output hold @ transmitter integrated delay	T_{holdT}	1.2	2.0	—	ns	8 pF		
	Data to clock input setup setup @ receiver integrated delay	T_{setupR}	1.0	2.0	—	ns	8 pF	Rx RGMII-ID	**Figure 48.21**
	Data to clock input setup hold @ receiver integrated delay	T_{holdR}	1.0	2.0	—	ns	8 pF		
	Clock cycle duration*2	T_{cyc}	7.2	8	8.8	ns	8 pF	—	—
	Duty cycle for gigabit	Duty_G	40*1	50	60*1	%	8 pF		
	Duty cycle for 10/100T	Duty_T	40*1	50	60*1	%	8 pF		
	Rise/fall time (20-80%)	T_r/T_f	—	—	0.75	ns	8 pF		

Note 1. Relaxed from regulation of RGMII.

Note 2. For 10Mbps and 100Mbps, T_{cyc} will scale to 400ns ±40ns and 40ns ±4ns respectively.

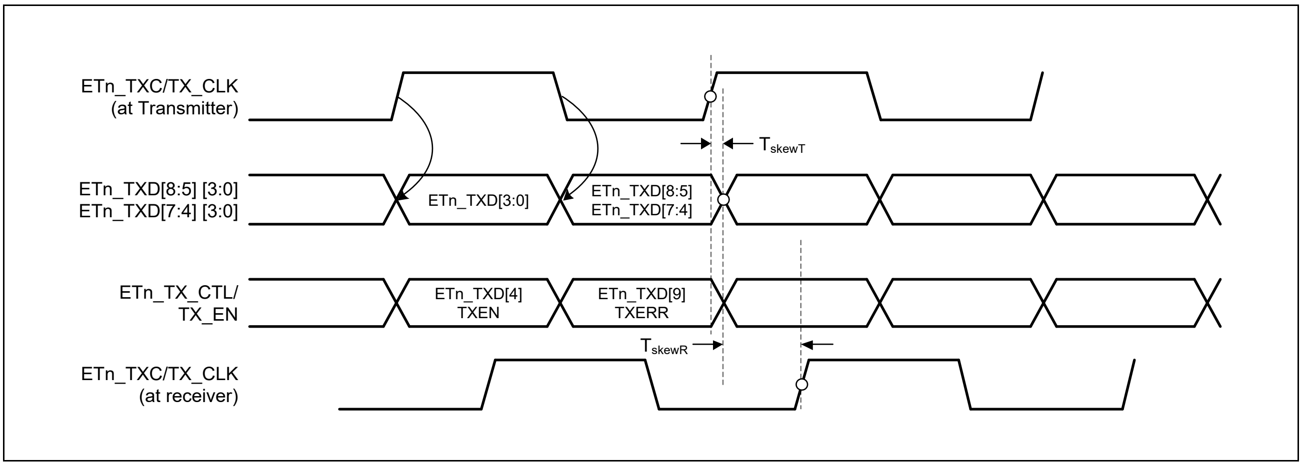


Figure 48.20 Multiplexing & Timing Diagram — RGMII (Transmitter)

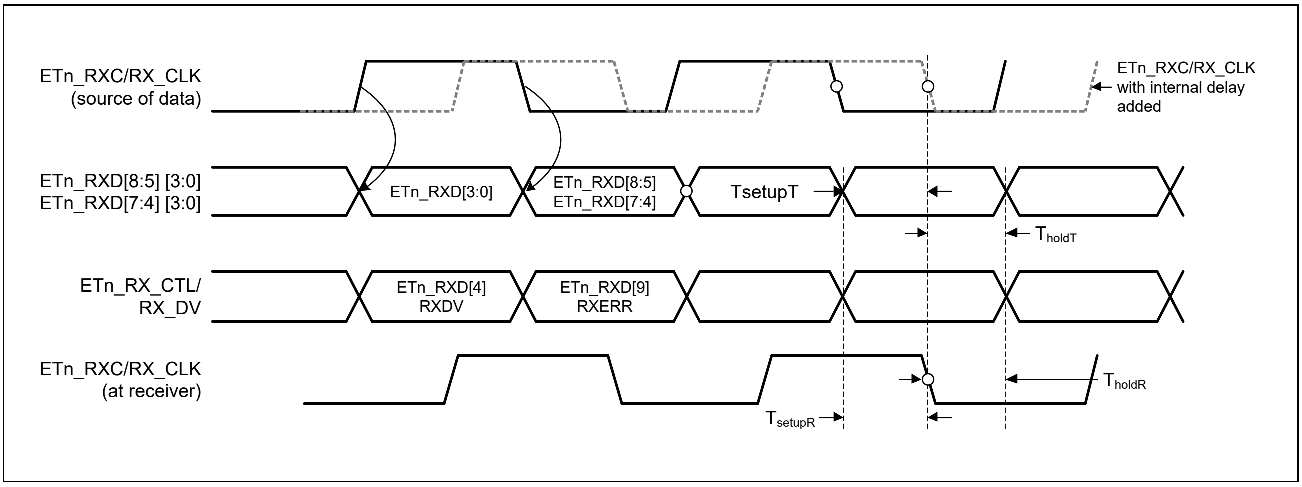


Figure 48.21 Multiplexing & Timing Diagram — RGMII-ID (Receiver)

48.5.8 JTAG Debugger Interface Access Timing

Table 48.33 Debugger IF Timing

Item	Symbol	Min.	Max.	Unit	Figures
TCK_SWCLK cycle time	t_{TCKcyc}	50	—	ns	**Figure 48.22**
TCK_SWCLK high pulse width	t_{TCKH}	20	—	ns	**Figure 48.23**
TCK_SWCLK low pulse width	t_{TCKL}	20	—	ns	
TDI setup time	t_{TDIS}	15	—	ns	
TDI hold time	t_{TDIH}	15	—	ns	
TMS_SWDIO setup time	t_{TMSS}	15	—	ns	
TMS_SWDIO hold time	t_{TMSH}	15	—	ns	
SWDIO delay time	t_{SWDO}	—	14	ns	
TDO delay time	t_{TDOD}	—	14	ns	
Capture register setup time	t_{CAPTS}	10	—	ns	**Figure 48.24**
Capture register hold time	t_{CAPTH}	10	—	ns	
Update register delay time	$t_{UPDATED}$	—	20	ns	

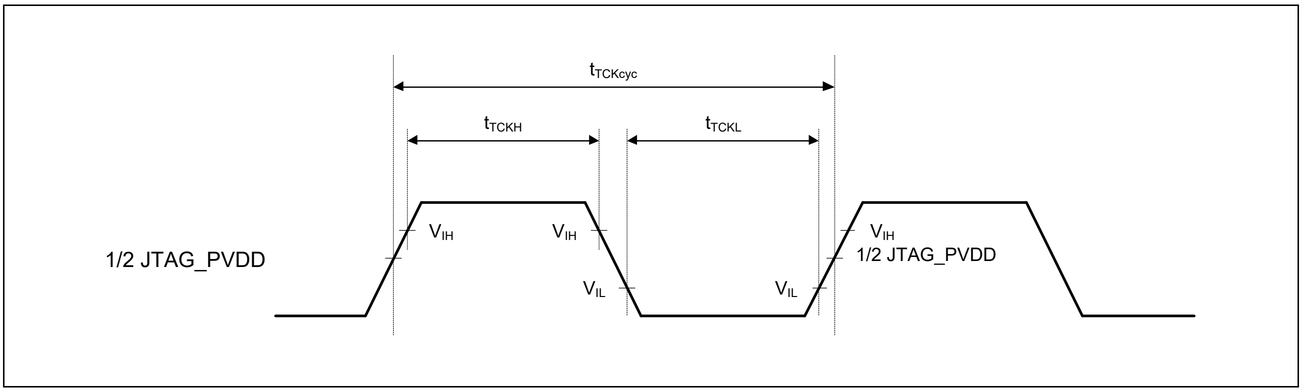


Figure 48.22 TCK_SWDCLK Input Timing

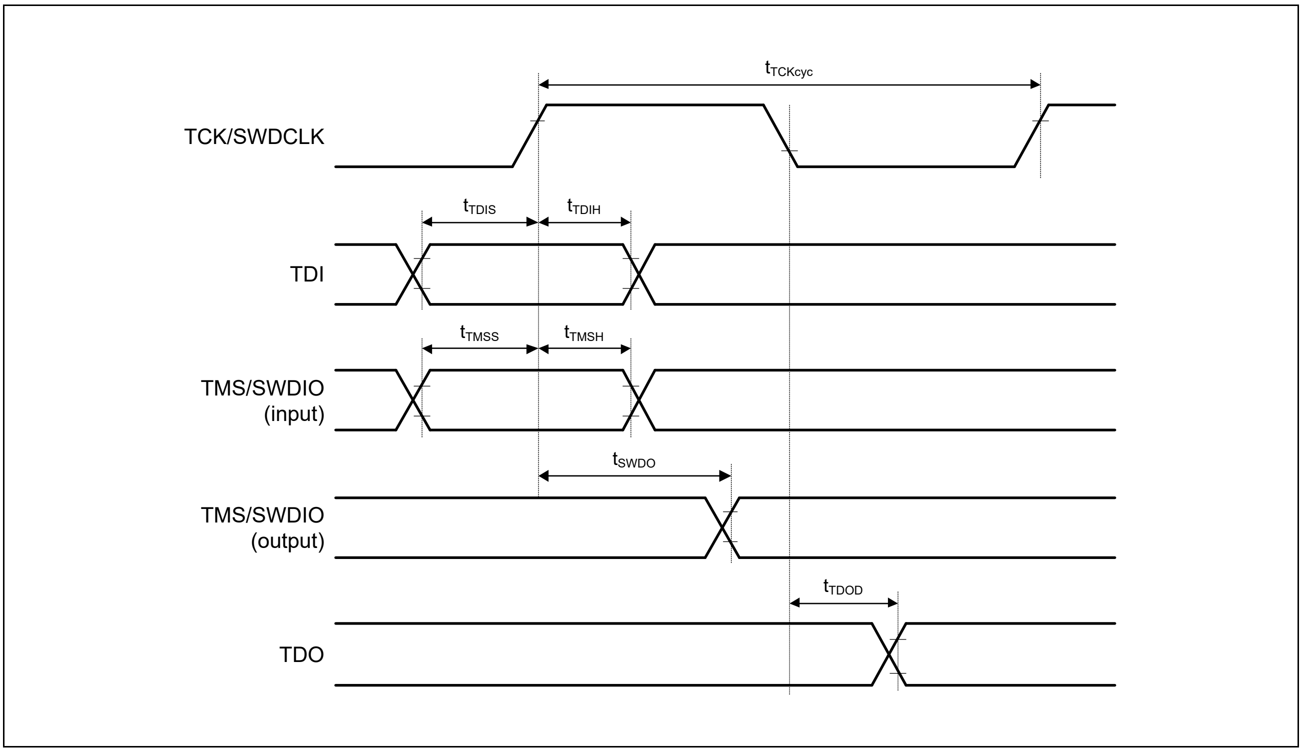


Figure 48.23 Data Transfer Timing

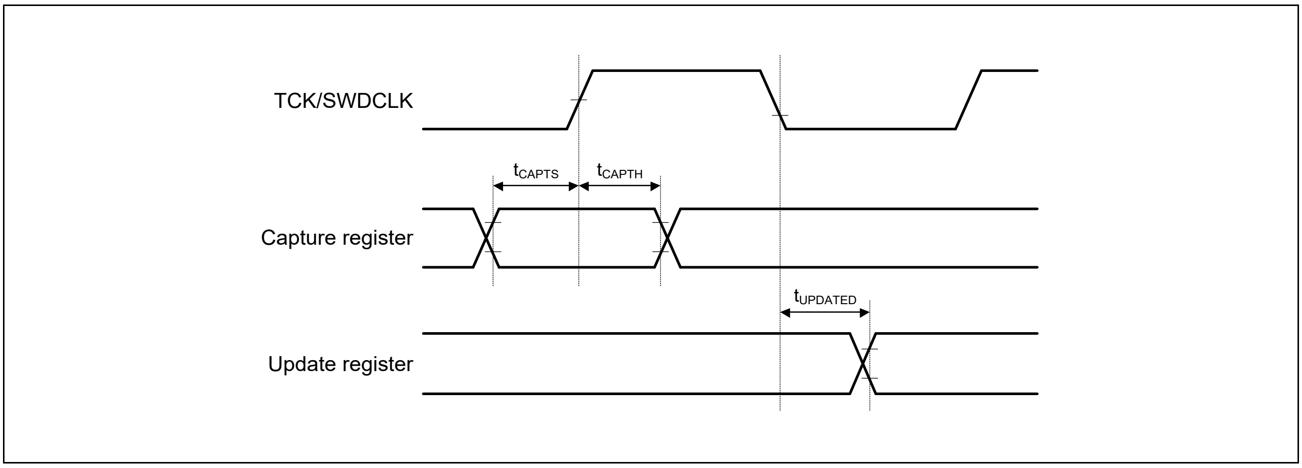


Figure 48.24 Boundary Scan Input/Output I/O Timing

48.5.9 SPI Multi I/O Bus Controller Access Timing

Table 48.34 SPI Multi I/O Bus Contoller Access Timing (1/2)

Item		Symbol	1.8 V (Serial flash/octal-SPI flash/hyper flash connected)		3.3 V (Serial flash connected)		Unit	Figures
			Min.	Max.	Min.	Max.		
Clock cycle		t_{SPBcyc}	10.0	—	15.0	—	ns	**Figure 48.25**, **Figure 48.29**
CLK high pulse width		t_{SPBWH}	0.45	0.55	0.45	0.55	t_{SPBcyc}	**Figure 48.25**, **Figure 48.29**
CLK low pulse width		t_{SPBWL}	0.45	0.55	0.45	0.55	t_{SPBcyc}	**Figure 48.25**, **Figure 48.29**
CLK rise time		t_{SPBR}	—	1.0	—	3.2	ns	**Figure 48.25**, **Figure 48.29**
CLK fall time		t_{SPBF}	—	1.0	—	3.2	ns	**Figure 48.25**, **Figure 48.29**
Data input setup time	QSPI0_SPCLK base point (SDR mode timing adjusted)	t_{SU}	6.7	—	7.5	—	ns	**Figure 48.26**
	QSPI0_SPCLK base point (DDR mode timing adjusted)		4.5	—	4.5	—	ns	**Figure 48.27**
	QSPI1_SSL base point (DQS base point)		−0.9[*2]	—	—	—	ns	**Figure 48.30**
Data input hold time	QSPI0_SPCLK base point (SDR mode timing adjusted)	t_H	0.5	—	0.5	—	ns	**Figure 48.26**
	QSPI0_SPCLK base point (DDR mode timing adjusted)		1.0	—	1.0	—	ns	**Figure 48.27**
	QSPI1_SSL base point (DQS base point)		2.69[*2]	—	—	—	ns	**Figure 48.30**
SSL setup time		t_{LEAD}	1.5 × t_{SPBcyc} − 3	8.5 × t_{SPBcyc} + 3	1.5 × t_{SPBcyc} − 3	8.5 × t_{SPBcyc} + 3	ns	**Figure 48.26**, **Figure 48.27**, **Figure 48.30**
SSL hold time		t_{LAG}	1 × t_{SPBcyc} − 3	8 × t_{SPBcyc} + 3	1 × t_{SPBcyc} − 3	8 × t_{SPBcyc} + 3	ns	**Figure 48.26**, **Figure 48.27**, **Figure 48.30**
Continuous transfer delay time		t_{TD}	1 × t_{SPBcyc} − 3	8 × t_{SPBcyc} + 3	1 × t_{SPBcyc} − 3	8 × t_{SPBcyc} + 3	ns	**Figure 48.26**, **Figure 48.27**, **Figure 48.30**
Data output delay time	SDR	t_{OD}	—	2.0	—	5.0	ns	**Figure 48.26**
	DDR		—	6.5[*2]	—	7.5[*3]	ns	**Figure 48.27**
	Hyper/octal		—	2.8	—	—	ns	**Figure 48.30**
Data output hold time	SDR	t_{OH}	−2.0	—	-5.0	—	ns	**Figure 48.26**
	DDR		1.0[*2]	—	2.1[*3]	—	ns	**Figure 48.27**
	Hyper/octal		0.9	—	—	—	ns	**Figure 48.30**
Skew of Clock to Data Strobe		t_{CKDS}	—	7.0	—	—	ns	**Figure 48.30**

Table 48.34 SPI Multi I/O Bus Conttoller Access Timing (2/2)

Item		Symbol	1.8 V (Serial flash/octal-SPI flash/hyper flash connected)		3.3 V (Serial flash connected)		Unit	Figures
			Min.	Max.	Min.	Max.		
Data output buffer off time	SDR	t_{BOFF}	—	2.0	—	3.0	ns	**Figure 48.28**
	DDR		—	2.0	—	3.0	ns	**Figure 48.28**
AC differential crossing voltage		V_{OX}	PVcc_SPI × 0.4	PVcc_SPI × 0.6	—	—	V	**Figure 48.29**

Note 1. Output load: 15 pF/driving ability: 12 mA

Note 2. QSPI0_SPCLK frequency: 100 MHz.

Note 3. QSPI0_SPCLK frequency: 66 MHz

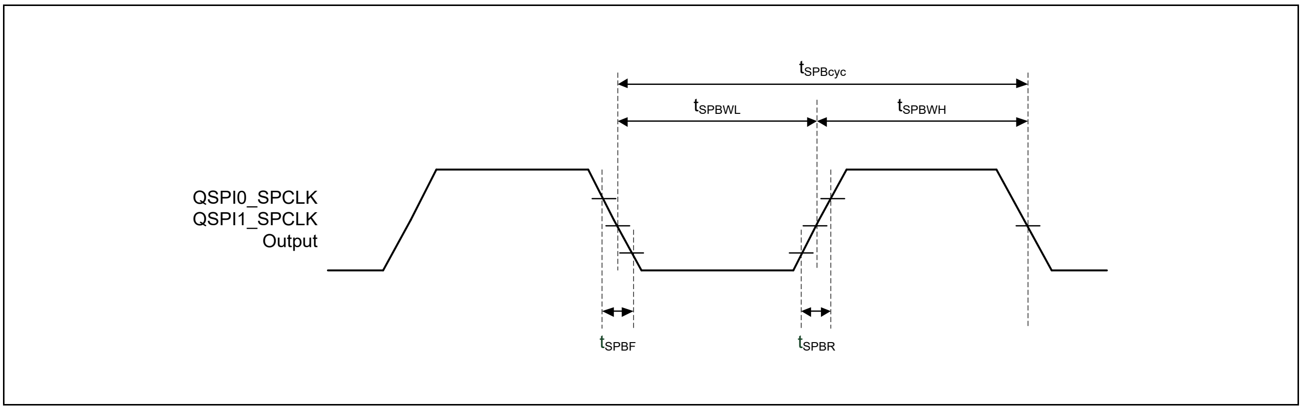


Figure 48.25 Clock Timing

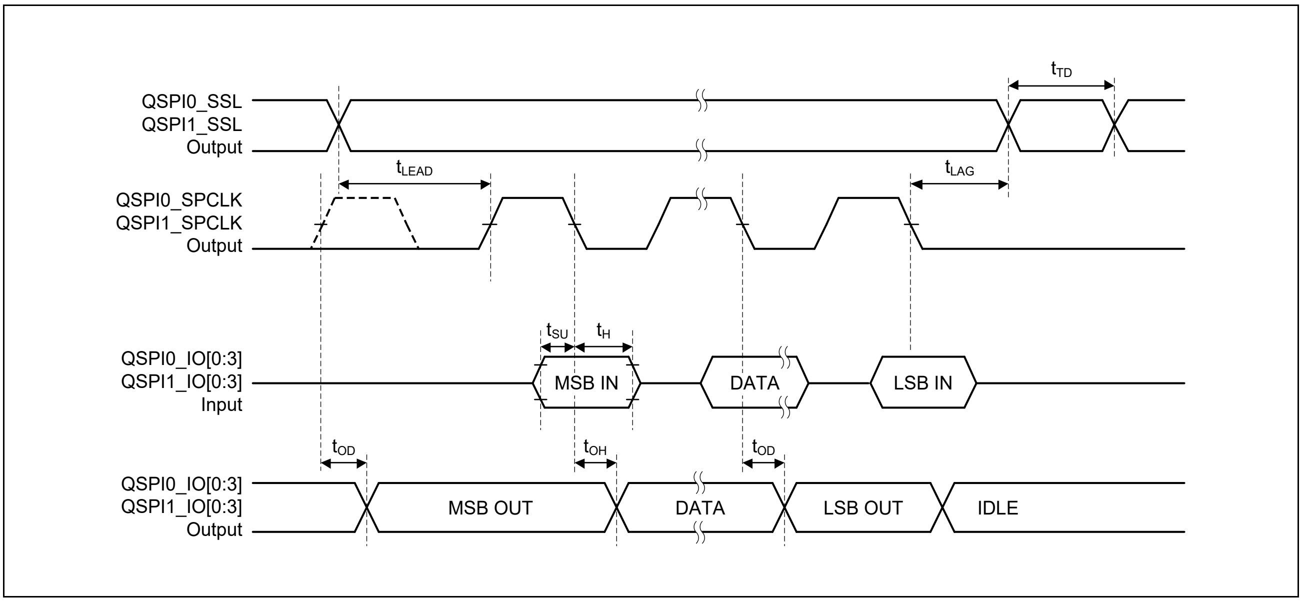


Figure 48.26 SDR Transfer Format Transmission and Reception Timing

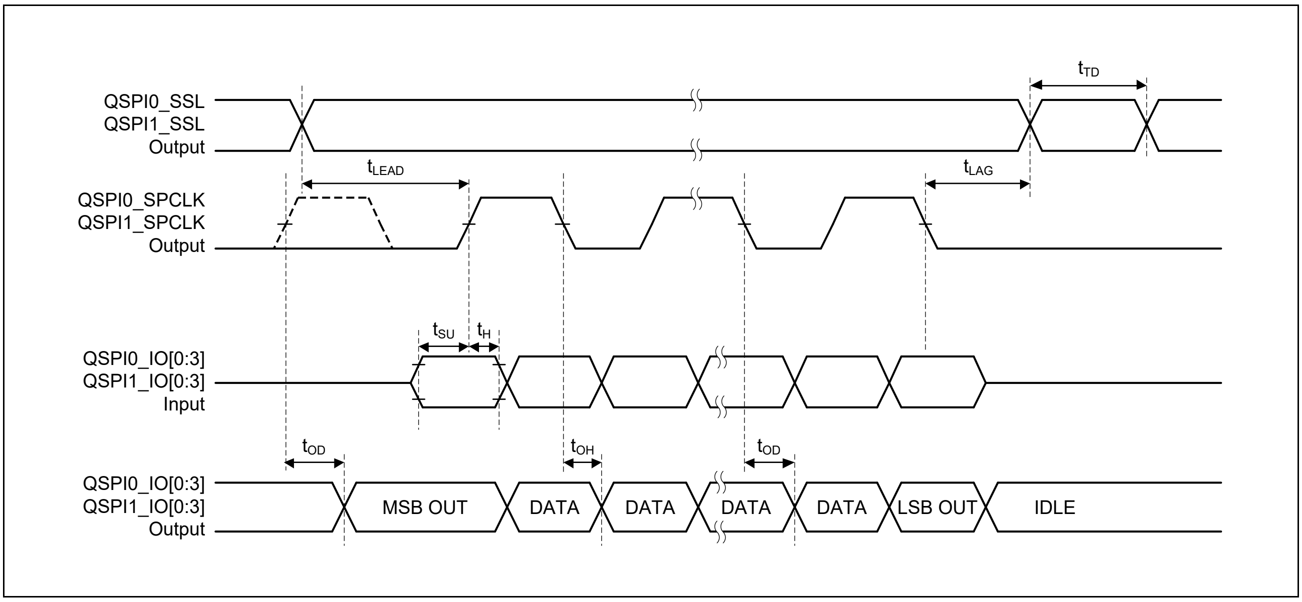


Figure 48.27 DDR Transfer Format Transmission and Reception Timing

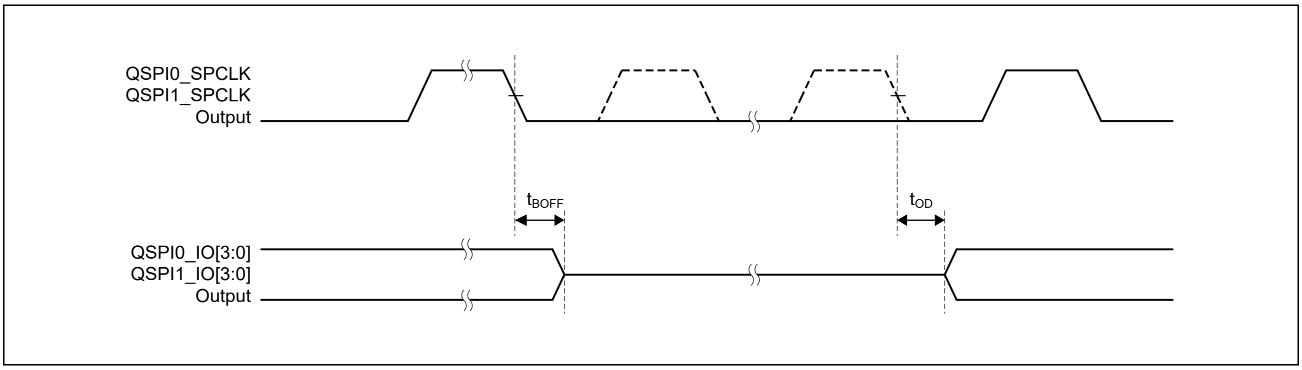


Figure 48.28 Timing for Switching the Buffers On and Off

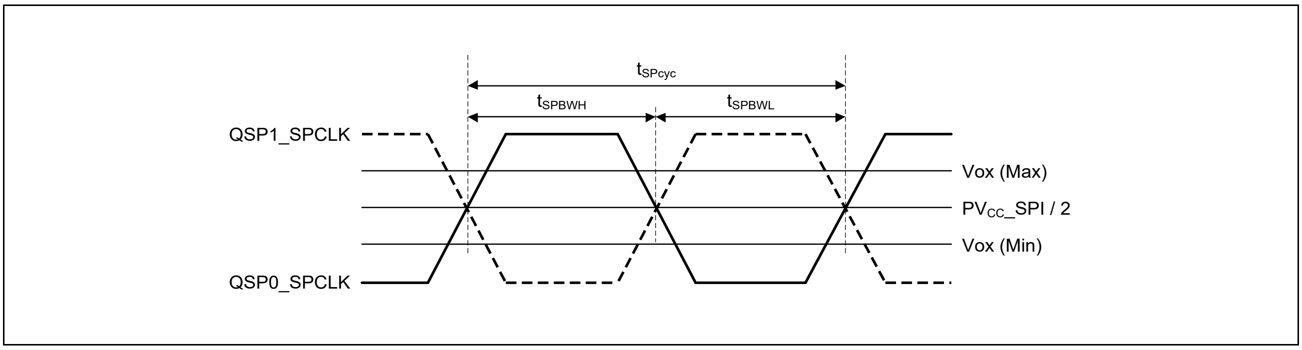


Figure 48.29 AC Differential Crossing Voltage

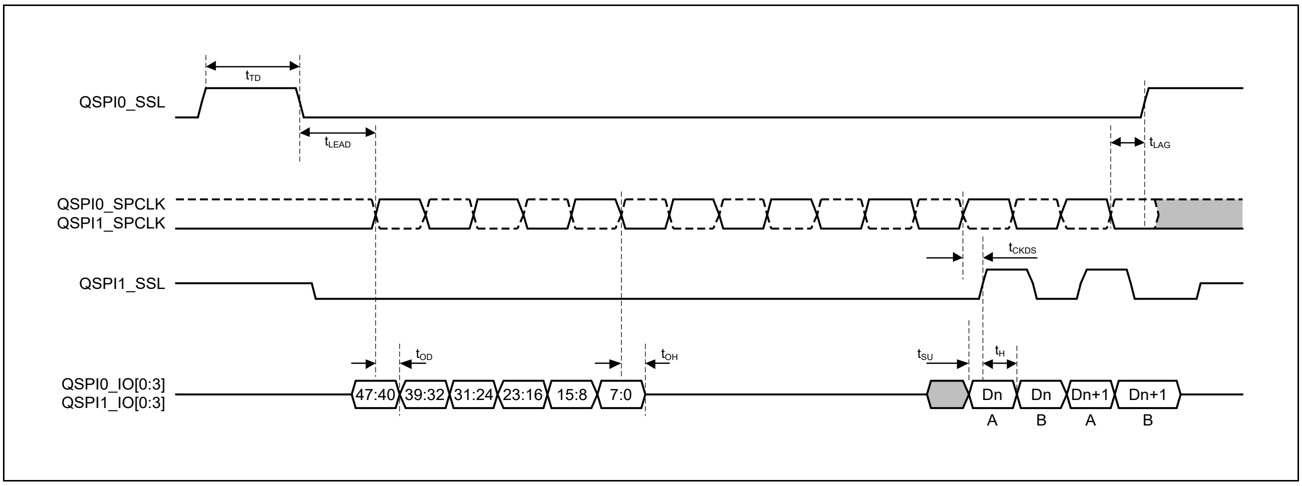


Figure 48.30 Transmit/Receive Timing with Octal-SPI Flash Memory or HyperFlash™ Connected

48.5.10 Control Signal Access Timing

Table 48.35 Control Signal Timing

Item	Symbol	Min.	Max.	Unit	Figures
PRST# pulse width	t_{RESW}	20	—	t_{cyc}*1	**Figure 48.31**
TRST# pulse width	t_{TRSW}	20	—	t_{cyc}*1	
NMI pulse width	t_{NMIW}	20	—	t_{cyc}*1	**Figure 48.33**
IRQ pulse width	t_{IRQW}	20	—	t_{cyc}*1	
TINT pulse width	t_{TINTW}	20	—	t_{cyc}*1	
PRST# input rise time	t_{RSr}	—	500	μs	**Figure 48.32**

Note 1. t_{cyc} = 41.666 ns (24 MHz)

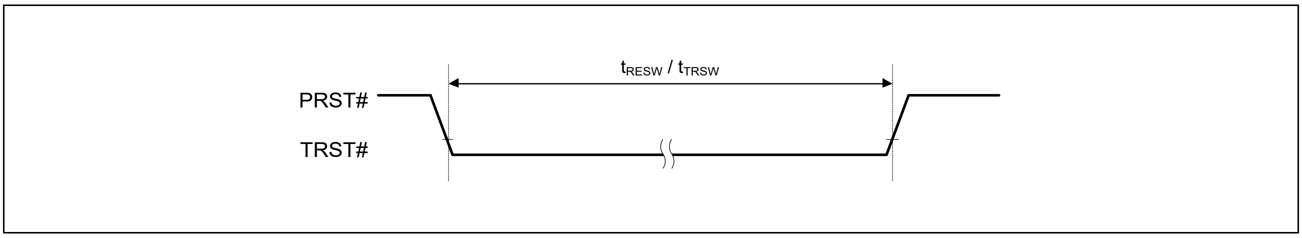


Figure 48.31 Reset Input Timing 1

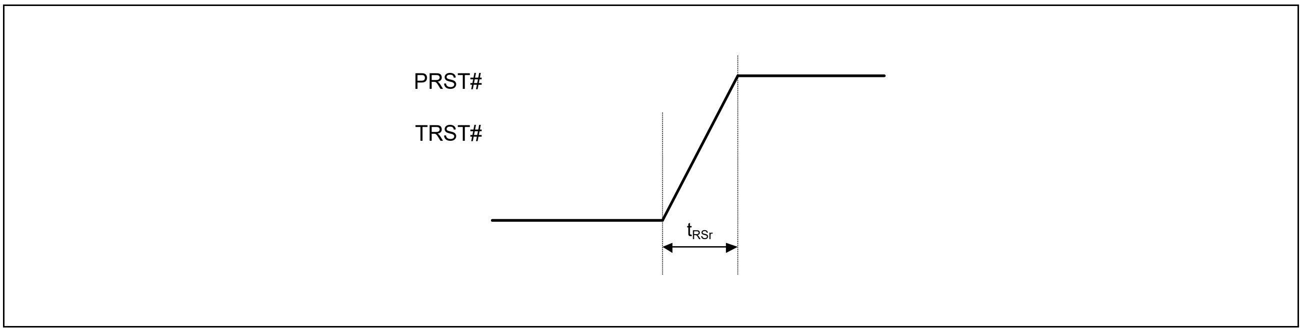
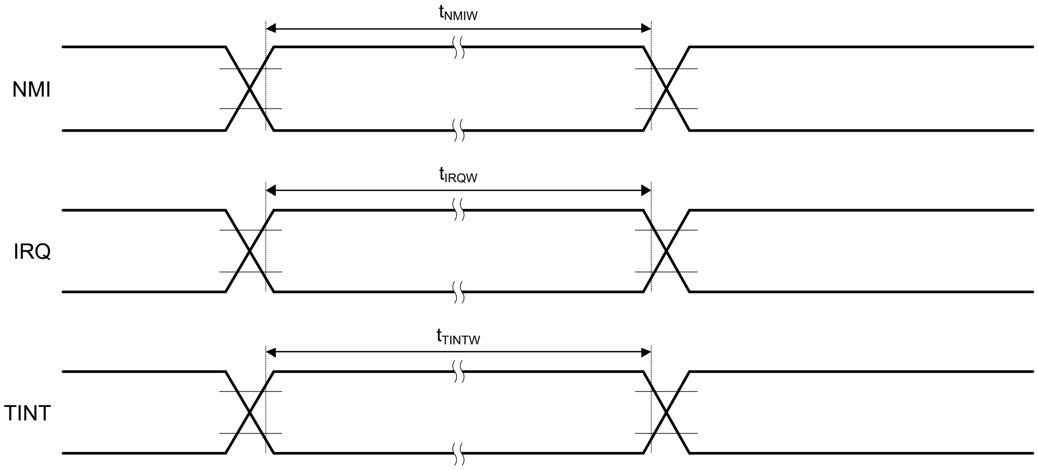


Figure 48.32 Reset Input Timing 2



Note: This specification “(Min. 20 t_{cycle})” is the min. pulse width the case that the digital noise filter is off. Refer to **Section 41, General Purpose Input Output Port (GPIO)** for Digital Noise Filter Settings in detail.

Figure 48.33 Interrupt Signal Input Timing

48.5.11 Serial Sound Interface (SSIF-2) Access Timing

Table 48.36 SSIF-2 Timing

Item	I/O	Symbol	Min.	Max.	Unit	Figures
Output clock cycle	Output	t_O	80	64000	ns	**Figure 48.34**
Input clock cycle	Input	t_I	80	—	ns	
Clock high	Bidirectional	t_{HC}	32	—	ns	
Clock low		t_{LC}	32	—	ns	
Clock rise time/clock fall time	Output	t_{RC}/t_{FC}	—	25	ns	
Setup time	Input	t_{SR}	25	—	ns	**Figure 48.35**, **Figure 48.36**, **Figure 48.37**
Hold time		t_{HR}	5	—	ns	
SILRCK output delay time	Output	t_{DTR}	−5	25	ns	
Data output delay time (Noise canceler not in use)		t_{DTR}	−5	25	ns	
Data output delay time (Noise canceler in use)		t_{DTR}	10	50	ns	

Note: AC access timing condition: drive ability 12mA, output load 30pF, slew rate = fast

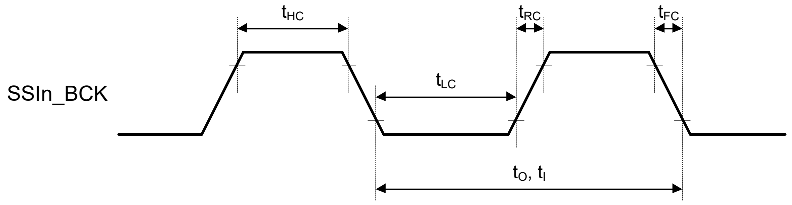


Figure 48.34 Bit Clock Input/Output Timing

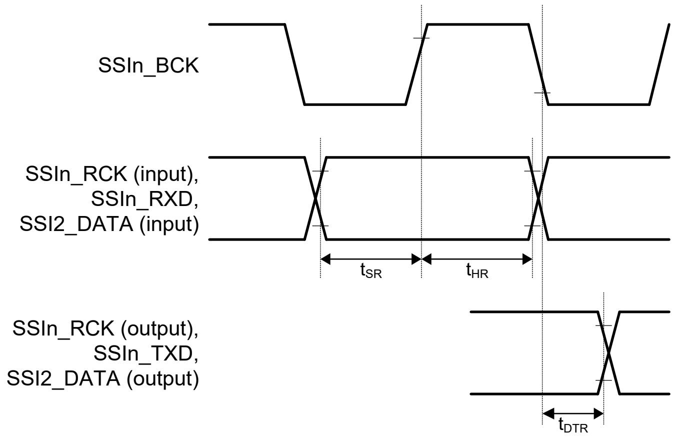


Figure 48.35 Transmission and Reception Timing (SSIBCK Falling Output)

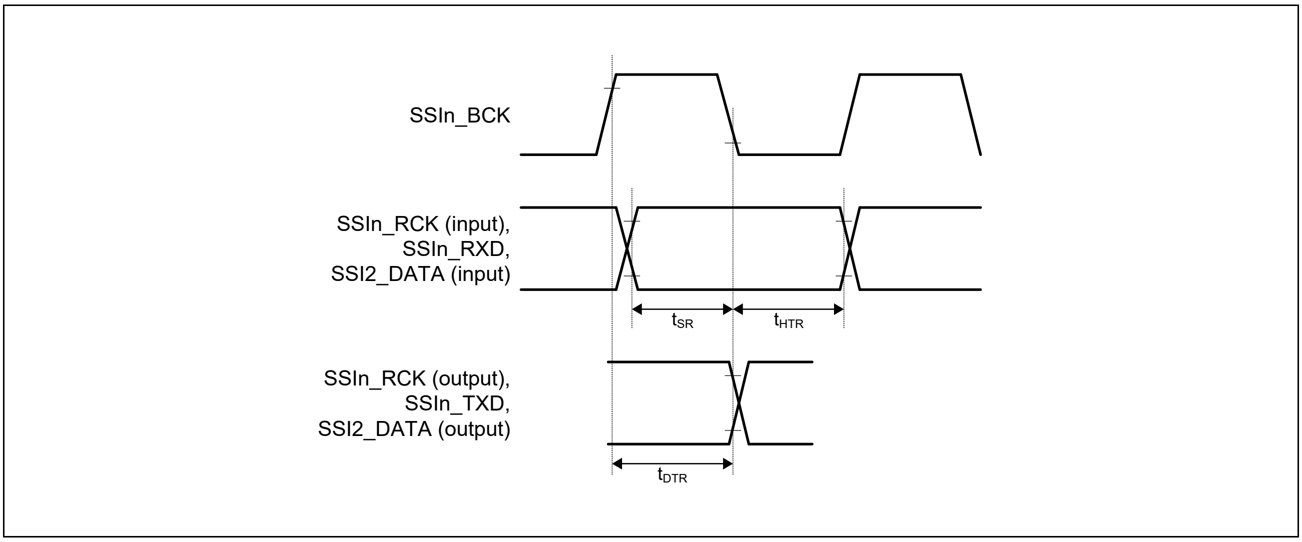
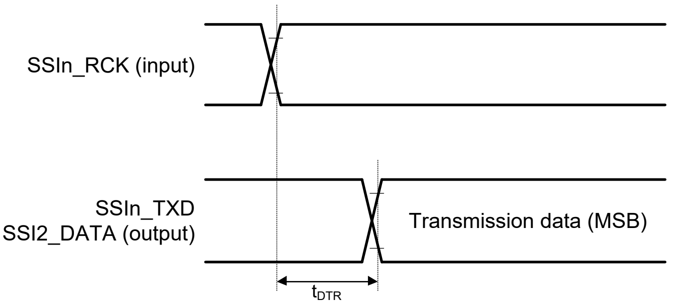


Figure 48.36 Transmission and Reception Timing (SSIBCK Rising Output)



Note: Output timing of the MSB when slave transmission satisfies the following condition a) or b)

a) SSICR.DEL = 1, SSICR.SDTA = 0

b) SSICR.DEL = 1, SSICR.SDTA = 1, without padding

Figure 48.37 Transmission Timing (Slave, in Synchronization with SSILRCK)

48.5.12 CAN-FD Interface Access Timing

Table 48.37 CAN-FD Interface Timing

		CAN		CAN-FD			
Item	Symbol	Min.	Max.	Min.	Max.	Unit	Figures
Internal delay time	t_{node}*1	—	100	—	75	ns	**Figure 48.38**
Transmission rate	—	—	1	—	4	Mbps	

Note: AC access timing condition: drive ability 12mA, output load 15pF, slew rate = fast

Note 1. Internal delay time (t_{node}) = Internal transfer delay time (t_{output}) + Internal receive delay time (t_{input})

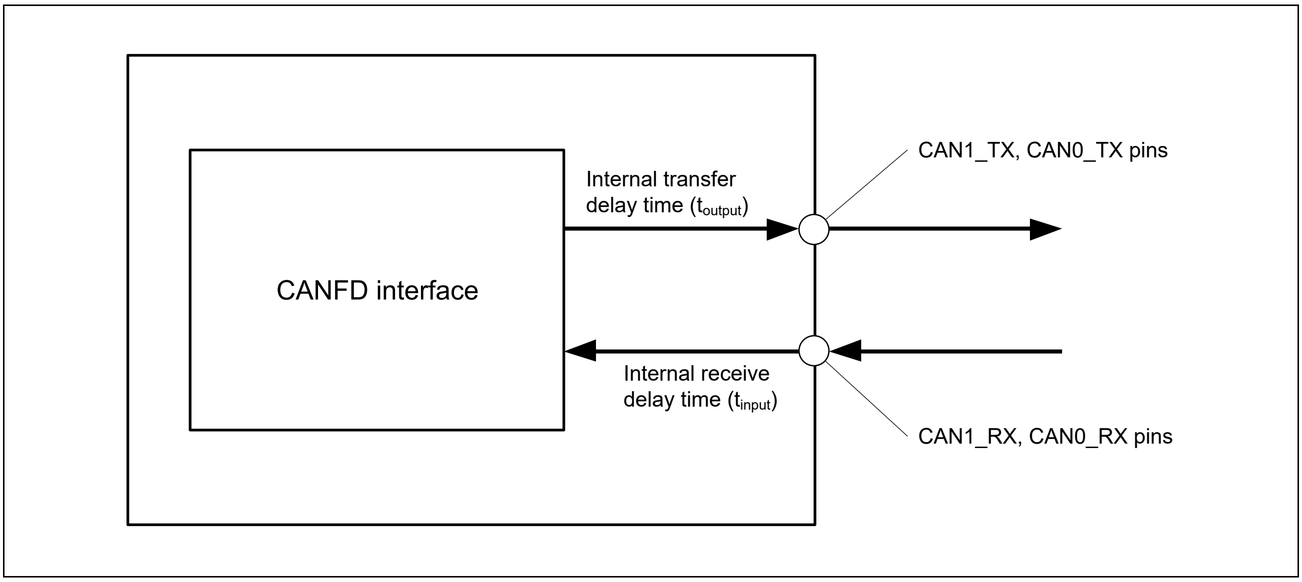


Figure 48.38 CAN-FD Interface Condition

48.5.13 Multi-Function Timer Pulse Unit 3 (MTU3a) Access Timing

Table 48.38 MTU3a Timing

Item			Symbol	Min.	Max.	Unit*1	Figures
MTU3a	Input capture input pulse width	Single-edge setting	t_{MTICW}	1.5	—	t_{p1cyc}*1	**Figure 48.39**
		Both-edge setting		2.5	—		
	Timer clock pulse width	Single-edge setting	t_{MTCKWH}, t_{MTCKWL}	1.5	—	t_{p1cyc}*1	**Figure 48.40**
		Both-edge setting		2.5	—		
		Phase counting mode		2.5	—		

Note: AC access timing condition: drive ability 4mA, output load 30pF, slew rate = fast

Note 1. t_{p1cyc} indicates peripheral clock means MTU_X_MCLK_MTU3 (P0ϕ).

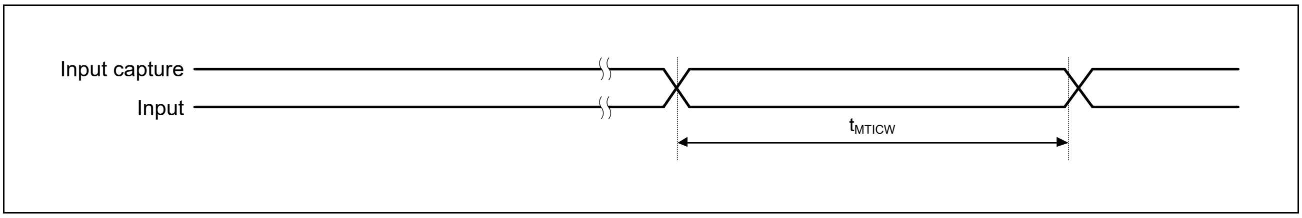


Figure 48.39 MTU3a Input Capture Input Timing

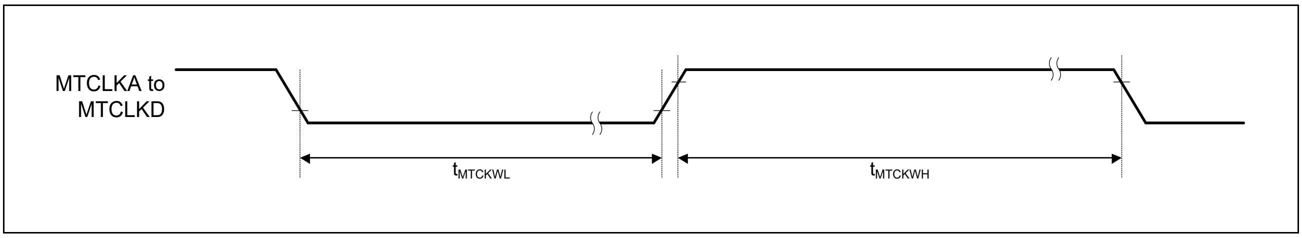


Figure 48.40 MTU3a Clock Input Timing

48.5.14 Port Output Enable 3 (POE3) Access Timing

Table 48.39 POE3 Timing

Item		Symbol	Min.	Max.	Unit	Figures
POE3	POEn# input pulse width	t_{POE3W}	1.5	—	t_{p1cy}*1	**Figure 48.41**

Note 1. t_{p1cyc} indicates peripheral clock means POE3_CLKM_POE (P0ϕ).

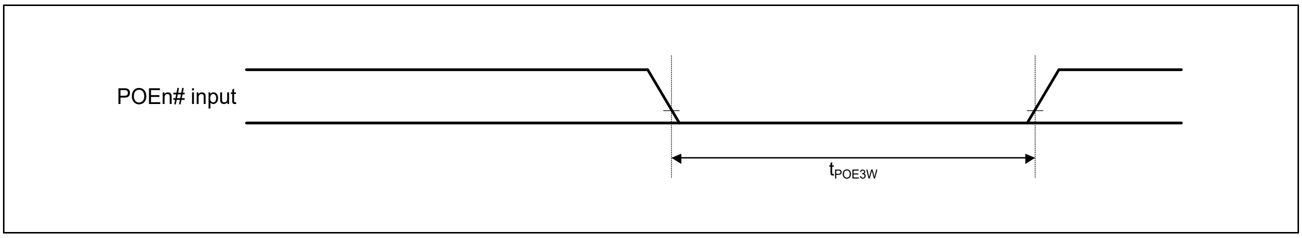


Figure 48.41 POEn# Input Pulse Timing

48.5.15 General PWM Timer (GPT) Access Timing

Table 48.40 GPT Timing

Item			Symbol	Min.	Max.	Unit	Figures
GPT	Input capture input pulse width	Single-edge setting	t_{GTICW}	1.5	—	t_{p1cyc}*1	**Figure 48.42**
		Both-edge setting		2.5	—		

Note: AC access timing condition: drive ability 4mA, output load 30pF, slew rate = fast

Note 1. t_{p1cyc} indicates peripheral clock means GPT_PCLK (P0ϕ).

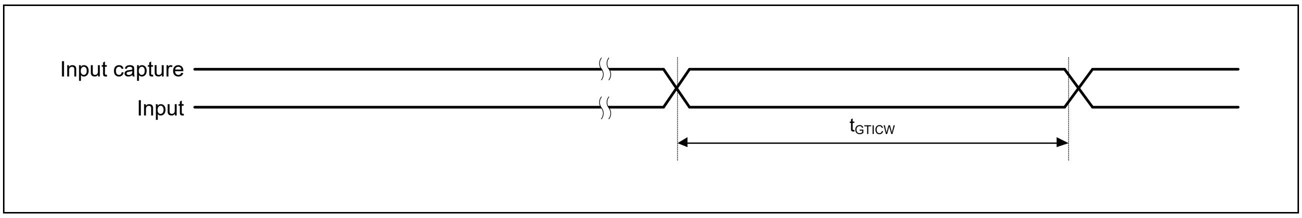


Figure 48.42 GPT Input Capture Input Timing

48.5.16 Port Output Enable for GPT (POEG) Access Timing

Table 48.41 POEG Timing

Item		Symbol	Min.	Max.	Unit	Figures
POEG	POEG input pulse width	t_{POEGW}	3	—	t_{p1cyc}*1	**Figure 48.43**

Note 1. t_{p1cyc} indicates peripheral clock means POEG_x_CLKP (P0ϕ) (x = A, B, C, D).

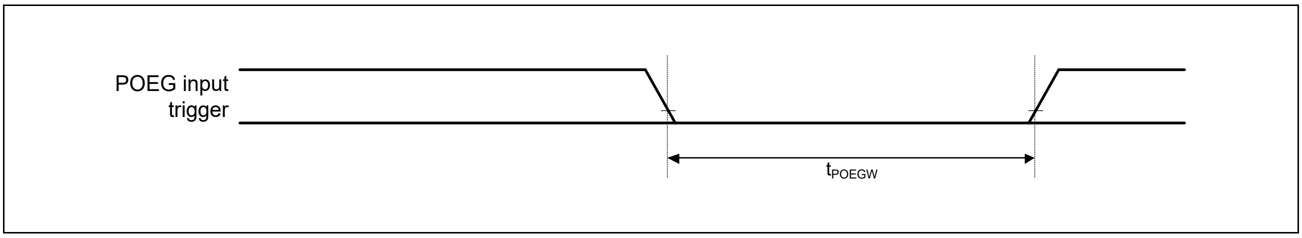


Figure 48.43 POEG Input Trigger Timing

48.5.17 I^{2}C Bus Interface Access Timing

Table 48.42 I^{2}C Bus Interface Timing

			Standard Mode (Sm)		Fast Mode (Fm)		Fast Mode Plus (Fm+)		
Item	Symbol	I/O	Min.	Max.	Min.	Max.	Min.	Max.	Unit
SCL clock frequency	f_{CLK}	I/O	0	100	0	400	0	1000	kHz
Bus free time (between stop and start condition)	t_{BUF}	I/O	4.7	—	1.3	—	0.5	—	μs
Hold time*1	$t_{HD:STA}$	I/O	4.0	—	0.6	—	0.26	—	μs
Low period of SCL clock	t_{LOW}	I/O	4.7	—	1.3	—	0.5	—	μs
High period of SCL clock	t_{HIGH}	I/O	4.0	—	0.6	—	0.26	—	μs
Setup time for start / restart condition	$t_{SU:STA}$	I/O	4.7	—	0.6	—	0.26	—	μs
Data hold time (I^{2}C bus device)	$t_{HD:DAT}$	I/O	0*2	—	0*2	—	0	—	μs
Data setup time	$t_{SU:DAT}$	I/O	250	—	100*3	—	50	—	ns
SDA and SCL signal rise time	t_R	Input	—	1000	20	300	—	120	ns
SDA and SCL signal fall time*3	t_F	Input	—	300	20 × (PV_{DD}/ 5.5 V)	300	20 × (PV_{DD}/ 5.5 V)	120	ns
		Output	—	300	20 × (PV_{DD}/ 5.5 V)*6	300*6	20 × (PV_{DD}/ 5.5 V)*7	120*7	ns
Setup time for STOP condition	$t_{SU:STO}$	I/O	4.0	—	0.6	—	0.26	—	μs
Capacitive load for each bus line	C_b	—	—	400*4	—	400*4	—	550*4	pF
Pulse width of spikes that must be suppressed by the input filter	t_{SP}	Input	—	—	0	50*5	0	50*5	ns

Note: In the above table and subsequently, SCL and SDA refer to the RIICnSCL and RIICnSDA signals, respectively.

Note: AC access timing condition: drive ability 4mA, output load 400pF, slew rate = slow

Note 1. The first clock pulse is generated on the SCL line after the start condition has been issued and the hold time has elapsed.

Note 2. This module requires a minimum of 300 ns hold time internally for the SDA signal to handle the period over which the falling edge of SCL has not reached a defined level (time until the CnSCL signal reaches V_{IL} (max.) from V_{IH} (min.)).

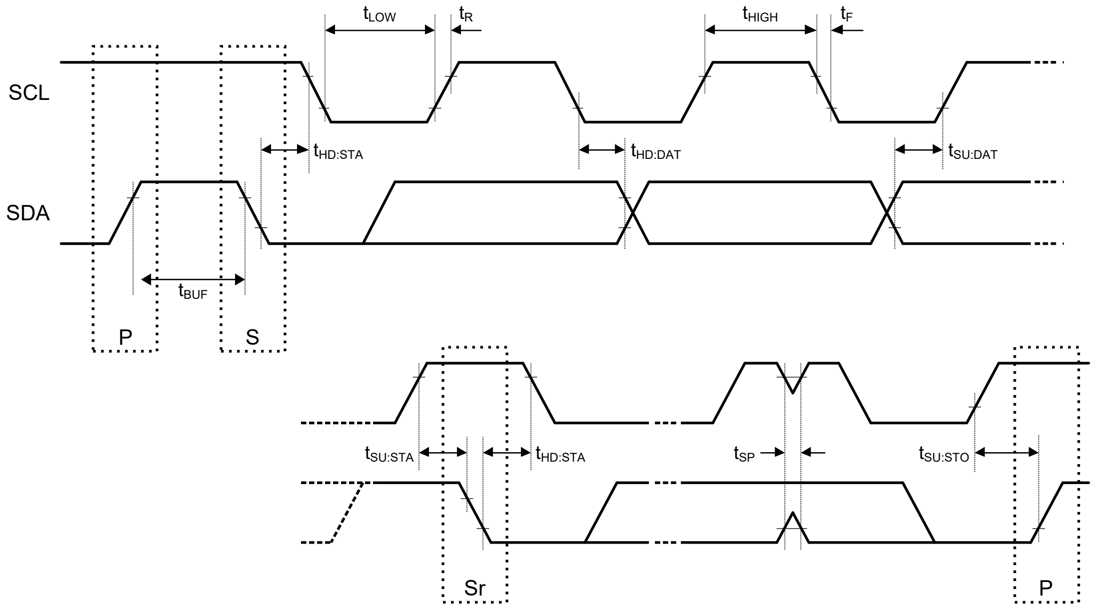
Note 3. The fast-mode I^{2}C bus device can be used in the standard mode I^{2}C bus system. In this case, the minimum value of the data setup time ($t_{SU:DAT}$ (min.) 250 [ns]) must be satisfied.
If the system does not extend the low period of SCL clock (t_{LOW}), this condition is automatically satisfied. If the system extends the low period of SCL clock (t_{LOW}), transmit the subsequent data bit to the SDA line before the SCL line is released (t_R (max.) + $t_{SU:DAT}$ (min.) = 1000 + 250 = 1250 [ns]: (standard mode I^{2}C bus specification)).

Note 4. Total capacitance of one bus line. The allowable maximum bus capacitance may differ from this specification, depending on the actual operating voltage and frequency of an application. For techniques to cope with a large bus capacitance, see the I^{2}C bus specification provided by NXP Semiconductors.

Note 5. Noise is removed by the analog and digital input filters. The level of noise reduction of the digital input filter is determined by the period of internal reference clock (IICϕ) and the NF[1:0] bits in RIICnMR3. For details, refer to **Section 26, I^{2}C Bus Interface**.

Note 6. External pull-up resistor is required 1077Ω to 1770Ω when using RIIC ch2 or RIIC ch3.

Note 7. External pull-up resistor is required 240Ω to 257Ω when using RIIC ch2 or RIIC ch3.



Note: P: Stop condition
S: Start condition
Sr: Start condition for retransmission

Figure 48.44 Input/Output Timing

48.5.18 Serial Communications Interface with FIFO (SCIFA) Access Timing

Table 48.43 SCIFA Timing

Item			Symbol	Min.	Max.	Unit	Figures
SCIFA	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{p1cyc}*1	**Figure 48.45**
		Clocked synchronous		12	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Input clock rise time		t_{SCKr}	—	5	ns	
	Input clock fall time		t_{SCKf}	—	5	ns	
	Output clock cycle	Asynchronous*2	t_{Scyc}	8	—	t_{p1cyc}*1	
		Clocked synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Output clock rise time		t_{SCKr}	—	9	ns	
	Output clock fall time		t_{SCKf}	—	9	ns	
	Transmit data delay time	Internal clock	t_{TXD}	−10	10	ns	**Figure 48.46**
		External clock		3 × t_{p1cyc}*1	4 × t_{p1cyc}*1 + 20		
	Receive data setup time	Internal clock	t_{RXS}	3 × t_{p1cyc}*1 + 20	—	ns	
		External clock		t_{p1cyc}*1 + 10	—		
	Receive data hold time	Internal clock	t_{RXH}	−3 × t_{p1cyc}*1	—	ns	
		External clock		2 × t_{p1cyc}*1 + 10	—		

Note: AC access timing condition: drive ability 12mA, output load 30pF, slew rate = fast

Note 1. t_{p1cyc} indicates peripheral clock means SCIFn_CLK_PCK (P0ϕ) (n = 0 to 4).

Note 2. When the SEMR.ABCS0 and SEMR.BGDM bits are set to 1.

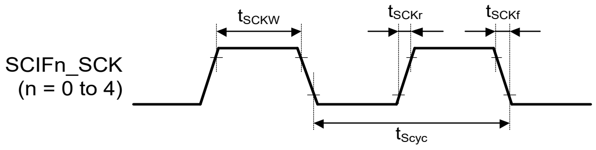
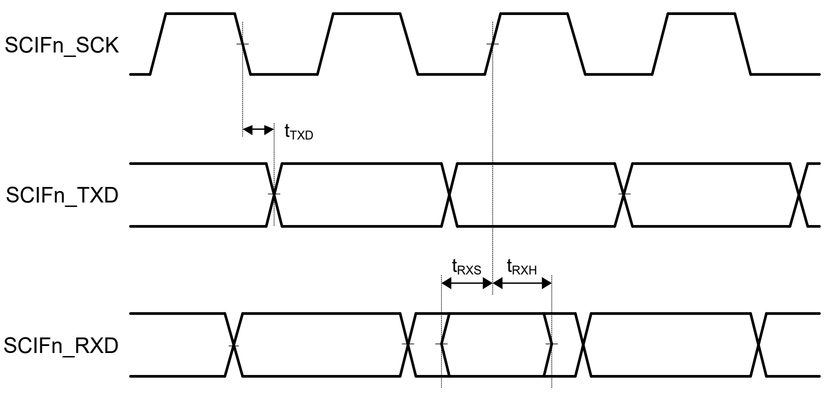


Figure 48.45 SCK Input Clock Timing



Note: n = 0 to 4

Figure 48.46 SCIFA Input/Output Timing in Clocked Synchronous Mode

48.5.19 Serial Communications Interface (SCIg) Access Timing

Table 48.44 SCIg Timing

Item			Symbol	Min.	Max.	Unit	Figures
SCIg	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{p1cyc}*1	**Figure 48.47**
		Clocked synchronous		6	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Input clock rise time		t_{SCKr}	—	5	ns	
	Input clock fall time		t_{SCKf}	—	5	ns	
	Output clock cycle	Asynchronous*2	t_{Scyc}	8	—	t_{p1cyc}*1	
		Clocked synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{p1cyc}*1	
	Output clock rise time		t_{SCKr}	—	5	ns	
	Output clock fall time		t_{SCKf}	—	5	ns	
	Transmit data delay time	Clocked synchronous	t_{TXD}	—	28	ns	**Figure 48.48**
	Receive data setup time	Clocked synchronous	t_{RXS}	15	—	ns	
	Receive data hold time	Clocked synchronous	t_{RXH}	5	—	ns	

Note: AC access timing condition: drive ability 12mA, output load 30pF, slew rate = fast

Note 1. t_{p1cyc} indicates peripheral clock means SCIn_CLKP (P0ϕ) (n = 0 to 1).

Note 2. When the SEMR.ABCS0 and SEMR.BGDM bits are set to 1.

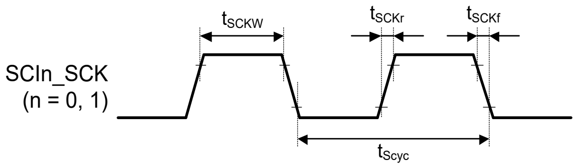
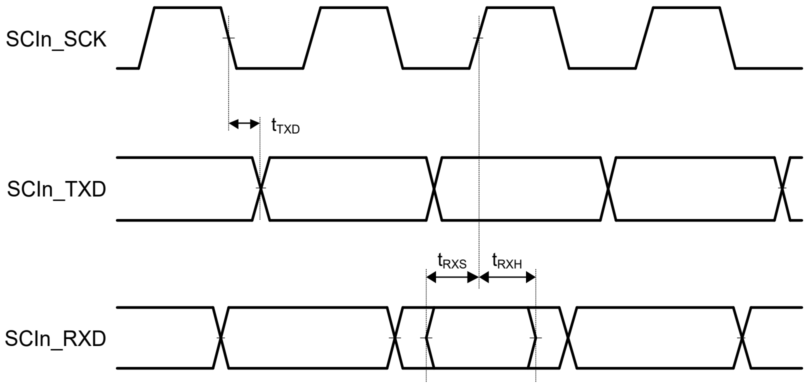


Figure 48.47 SCK Input Clock Timing



Note: n = 0, 1

Figure 48.48 SCIFA Input/Output Timing in Clocked Synchronous Mode

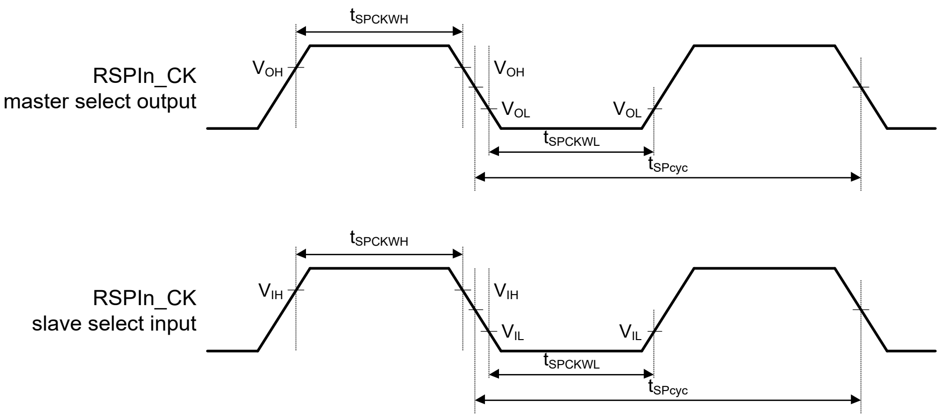
48.5.20 Renesas Serial Peripheral Interface (RSPI) Access Timing

Table 48.45 Renesas Serial Peripheral Interface Timing

Item		Symbol	Min.	Max.	Unit	Figure
RSPCK clock cycle	Master	t_{SPcyc}	2	4096	t_{p1cyc}*1	**Figure 48.49**
	Slave		8	4096		
RSPCK clock high pulse width	Master	t_{SPCKWH}	0.4	—	t_{SPcyc}*1	
	Slave		0.4	—		
RSPCK clock low pulse width	Master	t_{SPCKWL}	0.4	—	t_{SPcyc}*1	
	Slave		0.4	—		
Data input setup time	Master	t_{SU}	10	—	ns	**Figure 48.50** to **Figure 48.53**
	Slave		0	—	t_{p1cyc}*1	
Data input hold time	Master	t_H	0	—	ns	
	Slave		4	—	t_{p1cyc}*1	
SSL setup time	Master	t_{LEAD}	$1 \times t_{SPcyc} - 20$	$8 \times t_{SPcyc}$	ns	
	Slave		4	—	t_{p1cyc}*1	
SSL hold time	Master	t_{LAG}	$1 \times t_{SPcyc}$	$8 \times t_{SPcyc} + 20$	ns	
	Slave		4	—	t_{p1cyc}*1	
Data output delay time	Master	t_{OD}	—	19	ns	
	Slave		—	4	t_{p1cyc}*1	
Data output hold time	Master	t_{OH}	5	—	ns	
	Slave		2	—	t_{p1cyc}*1	
Continuous transmission delay time	Master	t_{TD}	$1 \times t_{SPcyc} + 2 \times t_{cyc}$	$8 \times t_{SPcyc} + 2 \times t_{cyc}$	ns	
	Slave		$4 \times t_{cyc}$	—		
Slave access time		t_{SA}	—	4	t_{p1cyc}*1	**Figure 48.52**, **Figure 48.53**
Slave out release time		t_{REL}	—	3	t_{p1cyc}*1	

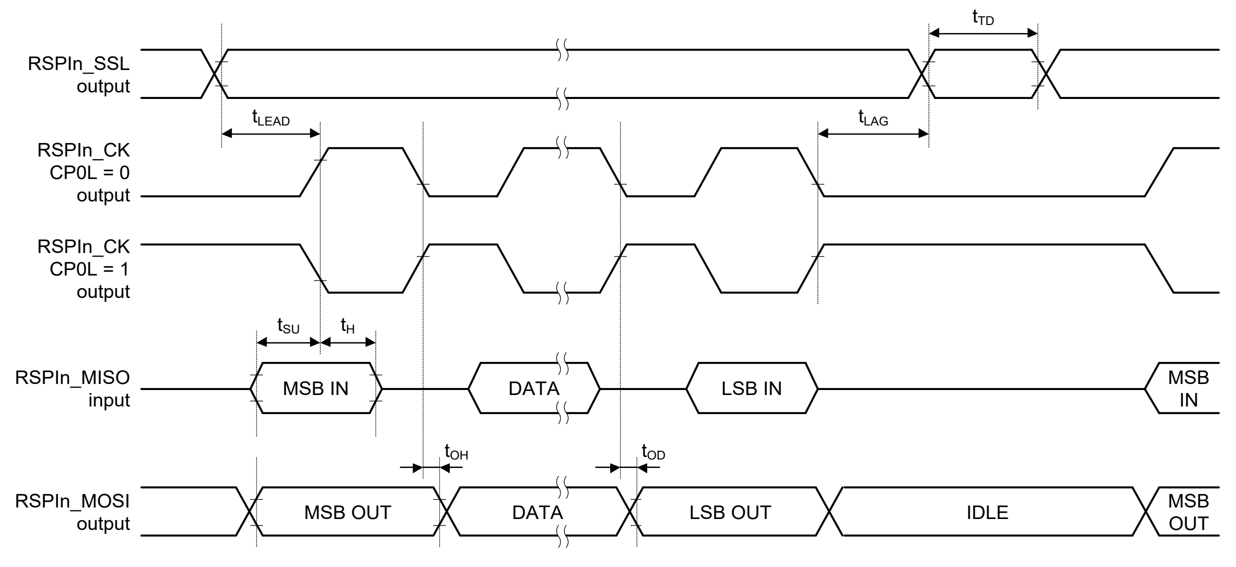
Note: AC access timing condition: drive ability 12mA, output load 30pF, slew rate = fast

Note 1. t_{p1cyc} indicates peripheral clock means RSPIn_CLKB (P0ϕ) (0 to 2).



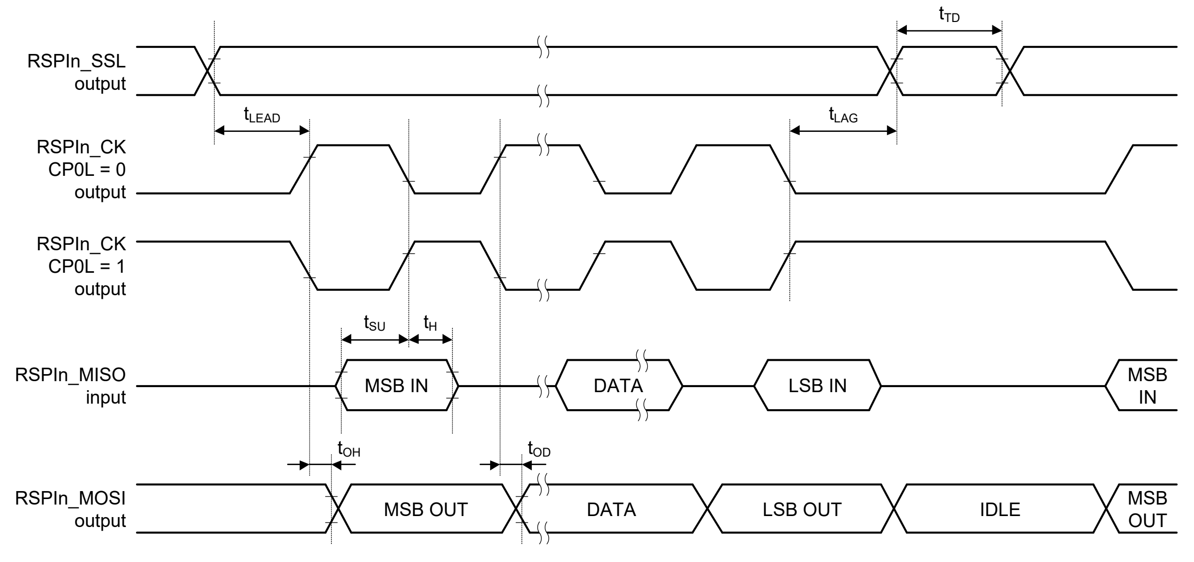
Note: n = 0 to 2

Figure 48.49 Clock Timing



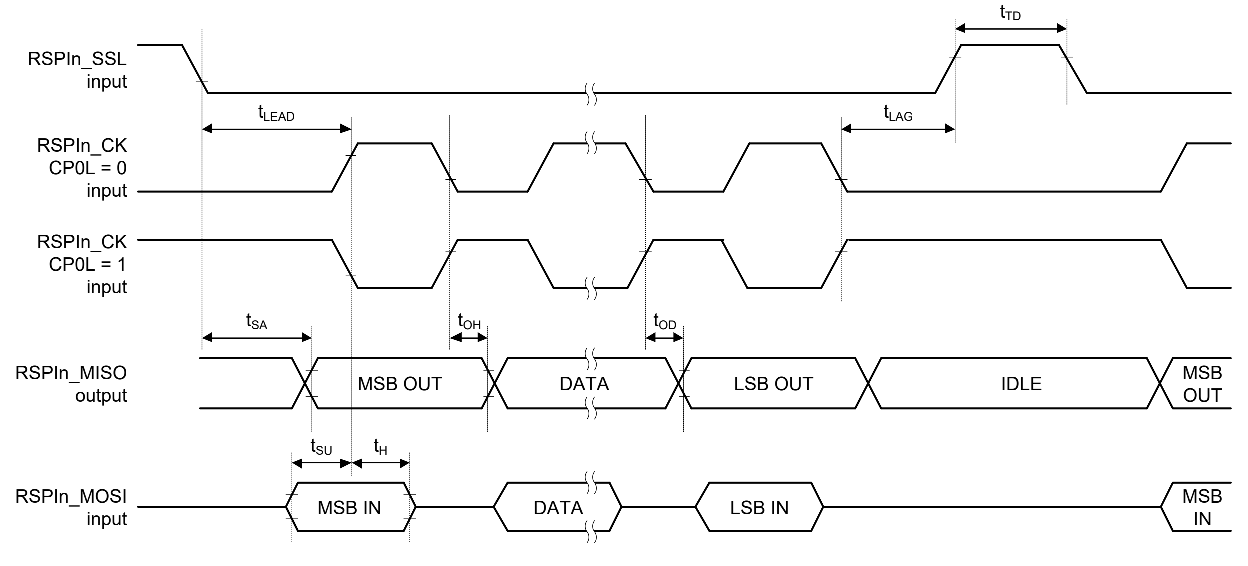
Note: n = 0 to 2

Figure 48.50 Transmission and Reception Timing (Master, CPHA = 0)



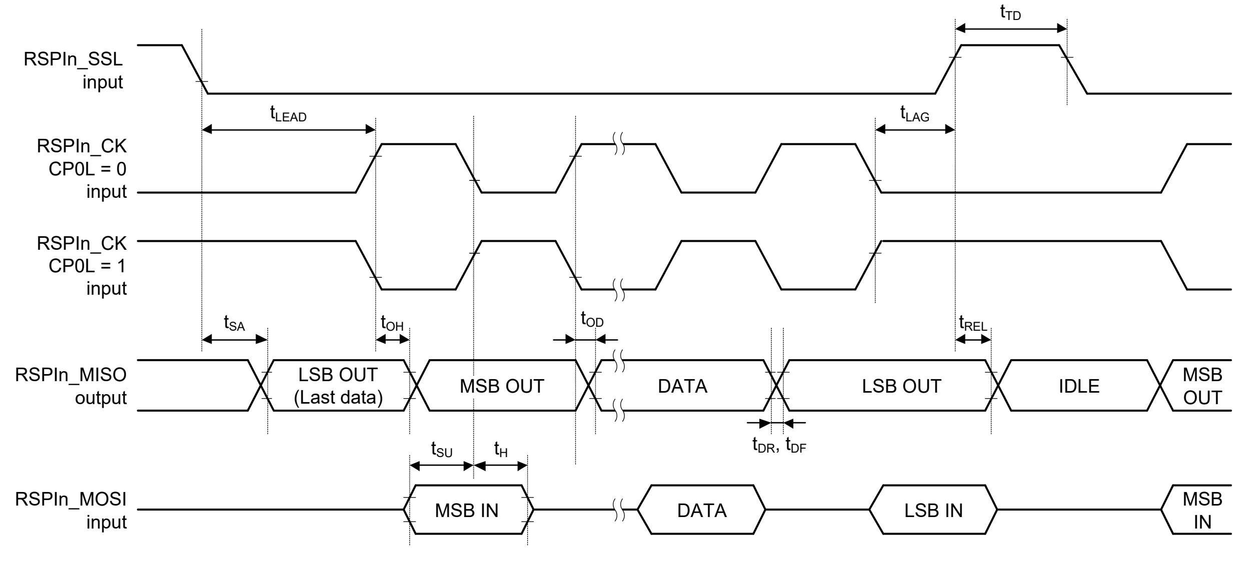
Note: n = 0 to 2

Figure 48.51 Transmission and Reception Timing (Master, CPHA = 1)



Note: n = 0 to 2

Figure 48.52 Transmission and Reception Timing (Slave, CPHA = 0)



Note: n = 0 to 2

Figure 48.53 Transmission and Reception Timing (Slave, CPHA = 1)

48.5.21 A/D Converter Access Timing

Table 48.46 A/D Converter Timing

Item	Symbol	Min.	Max.	Unit	Figures
ADC Trigger Input Pulse Width	t_{TRGW}	1.5*2		t_{P1cyc}*1	**Figure 48.54**

Note 1. t_{p1cyc} indicates peripheral clock means ADC_ADCLK (TSUϕ).

Note 2. When a noise filter in ADC is off.

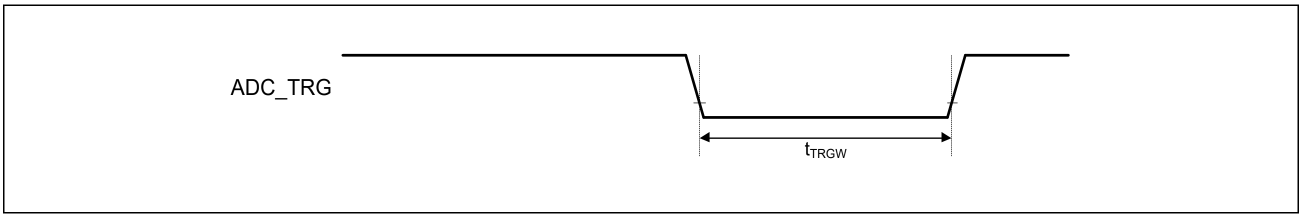


Figure 48.54 ADC Trigger Input Timing

48.5.22 Watchdog Timer Access Timing

Table 48.47 Watchdog Timer Timing

Item	Symbol	Min.	Max.	Unit	Figures
WDTOVF_PERROUT# Output Time	t_L	64	64	t_{P1cyc}*1	**Figure 48.55**

Note 1. t_{P1cyc} indicates peripheral clock means WDTn_CLK (OSCCLK) (n = 0 to 2).

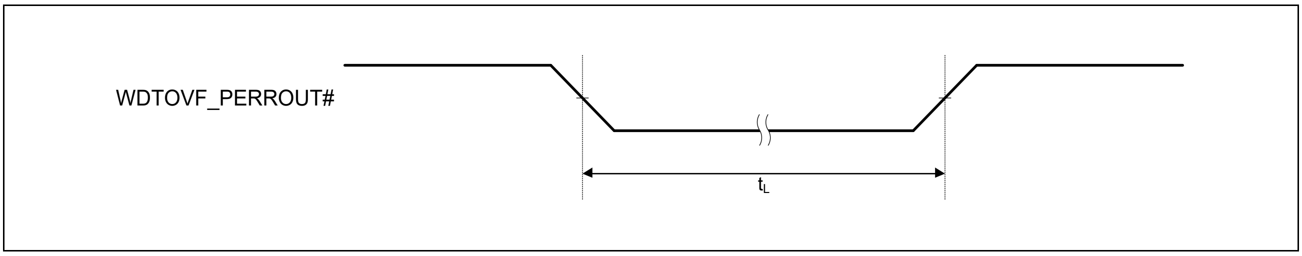


Figure 48.55 Watchdog Timer Output Timing

REVISION HISTORY	RZ/G2L Group, RZ/G2LC Group User’s Manual: Hardware

		Description	
Rev.	Date	Page	Summary
1.00	Sep 7, 2021	—	First edition issued
1.10	Apr 26, 2022	All	Binary notation, unified to ****b; Hexadecimal notation, unified to H’****
		3	Arm® Cortex®-A53 → Arm® Cortex®-M33
		1. Overview	
		71	1.2.2 CPU Peripheral Description of Clock Pulse Generator (CPG), modified (EXTAL → EXCLK)
		75	1.2.9 Sound Interface Serial Sound Interface (SSI), modified under [RZ/G2LC] (2 channels bidirectional serial transfer → 3 channels bidirectional serial transfer)
		77	1.2.11 Timer Description of Multi-function Timer Pulse Unit 3 (MTU3a), modified (P1ϕ → P0ϕ)
		7	1.2.12 Peripheral Module Serial Communication Interface with FIFO (SCIFA), modified under [RZ/G2LC] (3 channels → 4 channels)
		79	1.2.13 Security Trusted Secure IP (TSIP) [option]: ECC is added to Security algorism
		2. System CPU Cortex-A55	
		92	2.3.1 Cortex-A55 Sleep Mode The description of (3), modified (Cotex-A55 → Cortex-A55)
		3. System CPU Cortex-M33	
		100	3.4.4 Cortex-M33 Sleep Mode Note 3, added
		5. LSI Internal Bus	
		142 to 144	Table 5.1 Detailed Address Space The Space column of the following start addresses, modified (Start Address: H’0_4000_0000, H’0_2000_0000, H’0_1183_0000, H’0_1182_0000, H’0_1181_0000, H’0_1180_0000, H’0_1141_0000, H’0_1140_0000, H’0_1086_0000, H’0_1085_0000, H’0_1004_7000, H’0_1004_0000, H’0_0002_0000, H’0_0001_0000)
		139	5.1.2 Block Diagram of LSI Internal Bus The description of ACPU bus, modified
		140	Figure 5.1 Configuration of LSI Internal Bus: The blocks of MPU, Internal Memory, and Debug, modified
		146 to 151	Table 5.2 Detailed Address Space of Cortex-M33 The Space column of the following start addresses, modified (Start Address: H’9000_0000, H’8000_0000, H’7000_0000, H’6000_0000, H’5183_0000, H’5182_0000, H’5181_0000, H’5180_0000, H’5141_0000, H’5140_0000, H’5086_0000, H’5085_0000, H’5004_7000, H’5004_0000, H’4183_0000, H’4182_0000, H’4181_0000, H’4180_0000, H’4141_0000, H’4140_0000, H’4086_0000, H’4085_0000, H’4004_7000, H’4004_0000, H’3002_0000, H’3001_0000, H’2002_0000, H’2001_0000, H’1002_0000, H’1001_0000, H’0002_0000, H’0001_0000)
		152	5.3 Accessible Areas: The description, modified
		152, 153	Table 5.3 Accessible Areas The unit names in the Slave Unit and Master Unit columns, modified
		152	Table 5.3 Accessible Areas (1/2): Master Unit, modified (CRU [Video, Statistics] → CRU [Video])
		155	Table 5.4 List of Register Bits for Use in Re-setting of Security Attributes Read Access Control Bits of SYS_MSTACCCTL4, modified

Rev.	Date	Description Page	 Summary
1.10	Apr 26, 2022	6. System Controller (SYSC)	
		202	6.3.42 LSI Device ID Register (SYS_DEVID) The table of bits: Note, added to Description of the bit position 27 to 0
		7. Clock Pulse Generator (CPG)	
		230	Table 7.1 List of functions Specification of PLL control, modified
		236	7.2.2 Register classification: The description, modified
		280	7.2.4.27 Division Ratio Setting (PLL5) Register (CPG_PL5_SDIV) The Mode column in the table listed in Note, modified; Note 1, added
		479, 480	7.2.4.202 MSTOP Register REG0 (CPG_BUS_REG0_MSTOP) The bit chart: The bit names and R/W of the bits 16 and 0, modified; The table of bits: Bit Name, R/W, and Description of the bits 16 and 0, modified
		492	Table 7.12 Range of Reset Application (2/2) JTAG Interface of No.16, modified (Reset → Released)
		8. Interrupt Controller	
		540	8.6.4 IRQ Interrupt Type Selection Register (IITSR) The table of bits: Description of the bits 2n+1 to 2n, modified
		552	8.6.17 Bus Error Interrupt Status Control Register1 (BEISR1): The table of bits: The bit of BESTATn, modified (n → n-32)
		564	8.8.1 Precaution when use the peripheral modules which can initiate DMA Controller.: The description, modified (the referred section numbers, modified)
		9. DDR3L/DDR4 SDRAM Memory Controller (MEMC)	
		565	Table 9.1 DDR3L/DDR4 SDRAM Memory Controller (MEMC) features: Low power, deleted
		566	Figure 9.1 Block Diagram, modified (MC (cadence_mc_controller) → MC)
		10. On-chip RAM	
		569	The description, modified
		14. Direct Memory Access Controller	
		649	The section title, modified (14.4.16 MA Status EN Register → 14.4.16 DMA Status EN Register)
		652	The section title, modified (14.4.19 MA Status END Register → 14.4.19 DMA Status END Register)
		16. Multi-Function Timer Pulse Unit 3 (MTU3a)	
		932	Figure 16.121 Buffer Operation Timing (Compare Match) The waveform of TGRA, TGRB, modified
		933	Figure 16.123 Buffer Operation Timing (When TCNT Cleared) The waveforms of Buffer transfer signal and TGRA, TGRB, TGRE, modified
		21. Watchdog Timer (WDT)	
		1252	Table 21.6 WDT Reset Target by CPG_WDTRST_SEL Register CPG_WDTRST_SEL[3] and CPG_WDTRST_SEL[7], modified to Reserved
		22. Serial Communications Interface with FIFO (SCIFA)	
		1255	Table 22.1 Specifications of SCIFA Description of Channel, modified
		1292	Figure 22.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode Note 1, modified
		25. SPI Multi I/O Bus Controller	
		1460	Table 25.1 Pin Configuration Note 3, added

Rev.	Date	Description	
		Page	Summary
1.10	Apr 26, 2022	27. Serial Sound Interface (SSIF-2)	
		1628	Table 27.1 Features of SSIF-2 Description of Number of interfaces, modified
		1630	Figure 27.2 SSIF-2 Block Diagram AUDIO_CLK → AUDIO_CLK1; AUDIO_X1 → AUDIO_CLK2
		1672	Figure 27.27 Set timing and clear timing of TDE The waveform of TDE, modified
		1673	Figure 27.28 Set timing and clear timing of RDF The waveforms of Write point of receive FIFO, Read point of receive FIFO, Receive FIFO data full number, and RDF, modified
		1695	Figure 27.46 Continuation of the Data Communication State The waveform of Communication state, modified
		30. Gigabit Ethernet Interface	
		2077	Table 30.3 Register Base Address, modified
		2077	Table 30.4 Configuration of DMAC-related Registers (From H'000 to H'4FF) (1/2) Product Specific Register (PSR), deleted
		2082	30.4.1.1 DMAC Mode Register (CCC) The bit chart: R/W of the bit 20 (BOC), modified (R → R/W)
		2082	30.4.1.1 DMAC Mode Register (CCC) The table of bits: R/W of the bits 24 to 21, modified (R/W → R); Description of the bits 31 to 26, modified
		2085	30.4.1.3 Descriptor Base Address Load Request Register (DLR) The bit chart: Initial Value of the bit 4 (LBA4) and the bit 0 (LBA0), modified (0 → 1)
		2086 to 2087	30.4.1.4 DMAC Status Register (CSR) The table of bits: R/W of the bit 21 (TDUO) and the bit 16 (TPO), modified (R/W → R); R/W of the bit 10 (RCSI), the bit 9 (RDFDM), and the bit 8 (DTS), modified (— → R); R/W of the bits 3 to 0 (OPS), modified (R/W → R)
		—	30.4.1.7 Reserved, deleted
		2093	30.4.1.8 Reception Truncation Configuration Register I (RTC) The bit chart: Initial Value of the bit 14, modified (0 → 1); The table of bits: Initial Value of the bits 14 to 0 (MFL), modified (B'000 → H'7FFC)
		2097	30.4.1.11 Transmit Status Register (TSR) The bit chart: R/W of the bits 31 to 29, modified (R/W → R); The table of bits: Description of the bits 31 to 29, modified
		2098	30.4.1.12 MAC status FIFO Access Register (MFA) The bit chart: R/W of the bits 25 to 16 (MST) and the bits 9 to 0 (MSV), modified (R/W → R)
		2112	30.4.1.24 Receive Interrupt Enable Register 0 (RIE0) The table of bits: Initial Value of the bit 0 (FRS), modified (B'000 → 0b)
		2121	30.4.1.35 Receive Interrupt Control Register 3 (RIC3) The table of bits: Initial Value of the bit 0 (RDPE), modified (B'000 → 0b)
		2133	30.4.1.43 Interrupt Summary Status Register (ISS) The bit chart: R/W of the bits 31 to 14, the bits 12 to 10, and the bits 7, 6, 2, and 0, modified (R/W → R)
		2144	30.4.2.8 In-Band Status set register (CXR31) The bit chart: R/W of the bits 31 to 16, modified (R/W → R); The table of bits: Description of the bits 31 to 4, modified
		2150	30.4.2.12 PHY interface indicate register (CXR36) The table of bits: R/W of the bits 1 to 0 (STS_XMII), modified (R/W → R)
		2156	30.4.2.19 E-MAC operating mode register 2 (CXR2D) The table of bits: R/W of the bits 3 to 0, modified (R/W → R); The table of bits: Description of the bits 31 to 6 and 3 to 0, modified

Rev.	Date	Description	
		Page	Summary
1.10	Apr 26, 2022	2157	30.4.2.20 Software LINK status register (CXR2G) The table of bits: R/W of the bits 2 to 0, modified (R/W → R); The table of bits: Description of the bits 2 to 0, modified
		2159	30.4.2.24 TINT2 counter register (CXR41) The table of bits: Initial Value of the bits 15 to 0 (TINT2_CNT), modified (B’0 → H’0000)
		2170	30.4.2.39 Low Power Mode register 1 (LPTXMOD1) The table of bits: Initial Value of the bits 27 to 16 (GCYC_TCSTOP) and the bits 11 to 0 (MCYC_TCSTOP), modified (H’00 → H’000)
		2206	30.4.3.23 Lower MAC DA unicast address Condition register (CSFR13_L) The bit chart: R/W of the bits 31 to 16 (mac_da_uni), modified (R → R/W)
		2207	30.4.3.25 Lower MAC DA broadcast address Condition register (CSFR14_L) The bit chart: Initial Value of the bits 31 to 0 (mac_da_bro), modified (0 → 1)
		2208	30.4.3.26 Upper MAC DA multicast address Condition register i (i = 0 to 19) (CSFR15_U_i) The table of bits: Initial Value of the bits 15 to 0 (mac_da_mul_i), modified (H’0010 → H’0100)
		2208	30.4.3.27 Lower MAC DA multicast address Condition register i (i = 0 to 19) (CSFR15_L_i) The table of bits: R/W of the bits 31 to 0, modified (R → R/W)
		2212	30.4.3.29 IPv6 analysis Protocol Condition register 0 (CSFR16_0) The bit chart: Initial Value of the bits 20, 16, 10, and 9, modified (0 → 1)
		2213	30.4.3.30 IPv6 analysis Protocol Condition register 1 (CSFR16_1) The table of bits: Initial Value of the bits 31 to 24 (v6_ana_protocol_7), modified (All 0 → H’ 3A); Initial Value of the bits 23 to 16 (v6_ana_protocol_6), modified (B’0 → H’33); Initial Value of the bits 15 to 8 (v6_ana_protocol_5), modified (All 0 → H’2C); Initial Value of the bits 7 to 0 (v6_ana_protocol_4), modified (B’0 → H’2B)
		2222	30.4.3.35 Auto Response Configuration register1 (CSFR30) The bit chart: Initial Value of the bit 6, modified (1 → 0); The table of bits: Initial Value of the bits 16, 14, 13, 12, 11, 9, 8, 5, 4, 2, and 0, modified (B’ 0 → 1b)
		2224	30.4.3.36 Auto Response Configuration register2 (CSFR31) The table of bits: R/W of the bits 31 to 24 (na_tc), modified (R → R/W); R/W of the bits 23 to 20, modified (R/W → R); Description of the bits 23 to 20, modified
		32. USB2.0	
		2373	The section title, modified (32.5 Resume form Direct power down mode → 32.5 Usage Notes)
		32A. USB 2.0 Host Module	
		2454	(5) Line Control Port 1 Register The bit chart: Bit Name of the bit 19, modified (DP_RPD → DPRPD_EN)
		32B. USB 2.0 Function Module	
		2524	32B.2.14.1 DCP Configuration Register [DCPCFG] <Address: 05CH> The bit chart: R/W of the bits 8 (CNTMD) and 7 (SHTNAK), modified (— → R/W)
		2542	32B.2.16.1 PIPE9 Control Register [PIPE9CTR] <Address: 080H>, deleted
		33. LCDC	
		2659	Figure 33.1 Block Diagram ufcpvhtop0, vsp1z_du_g2l, and lif2dif, deleted
		2676	Table 33.8 DU Register Configuration DU Module Control Register 1, DU PBUF Control Register 1, and DU PBUF Control Register 2, added
		2793	33.3.3.8 DU Display I/F Timing Register 3 (DU_DITR3) The table of bits: Bit Name of the bits 28 to 16, modified (VACTIVE → HACTIVE)
		2796	33.3.3.11 DU Module Control Register 1 (DU_MCR1), added
		2798	33.3.3.13 DU PBUF Control Register 1 (DU_PBCR1), added
		2799	33.3.3.14 DU PBUF Control Register 2 (DU_PBCR2), added

Rev.	Date	Description	
		Page	Summary
1.10	Apr 26, 2022	35. Camera Data Receiving Unit (CRU)	
		2950	35.1.5 Register Configuration The table, modified
		2950, 2951	Table 35.4 Register Configuration of MIPI CSI-2 LINK The Address column, modified
		2952	35.1.6.1 Module Configuration Register (CSI2nMCG) The bit chart: Initial Value of the bit 10, modified (0 → 1); The table of bits: Initial Value of the bits 11 to 8 (SDLN), modified (H'0 → H'4)
		2969	35.1.6.17 Virtual Channel (M) Status Register (CSI2nVCST(M)) The bit chart: R/W of the bits 31 to 16, modified (R/W → R)
		2985	35.1.6.30 D-PHY Timing Register 0 (CSIDPHYTIM0) The bit chart: Initial Value of the bit 16 (T_INIT), modified (1 → 0); Initial Value of the bits 13, 10 to 8, and 4 (T_INIT), modified (0 → 1); The table of bits: Initial Value of the bits 18 to 0 (T_INIT), modified (H'10000 → H'0_2710)
		2990	(2) Generic Short Packet Data Type Code: The description, modified
		2991	Figure 35.5 Generic Short Packet Receive Processing, modified
		2997	35.2.1.1 Features Image format conversion (YCbCr422 → YCbCr420): The description, added
		3007	35.2.3.3 CRU Interrupt Status Register (CRUnINTS) The table of bits: Description of the bits 20 (CES), 19 (WIS), 18 (SIS), 17 (EFS), 16 (SFS), 3 (FEOVWS), 2 (DECES), 1 (SLVES), and 0 (FOS), modified
		3027	35.2.3.34 CRU Image Processing Main Control Register (ICnMC) The bit chart: Initial Value of the bit 13 (CLP), modified (0 → 1); The table of bits: Initial Value of the bits 13 to 12 (CLP), modified (B'01 → B'11)
		3041	35.2.3.47 CRU Output Image Format Register (ICnDMR) The table of bits: Description of the bits 6 to 4 (YCMODE), modified
		3059, 3060	Table 35.12 Input Format and Image Processing The columns with "—" in the column heading, deleted
		3081	Figure 35.33 Reception Start Flow for the MIPI CSI-2 Input Note 1 and Note 2, added
		3082	Figure 35.34 Initializing the D-PHY (MIPI CSI-2 Input) DPHYTIM0 → CSIDPHYTIM0; DPHYTIM1 → CSIDPHYTIM1; DPHYCTRL0 → CSIDPHYCTRL0; The description in the third box from the top, modified (DPHYTIM0 and DPHYTIM1 → CSIDPHYTIM0; CSIDPHYTIM1 and CSIDPHYSKW0)
		3084	Figure 35.36 Initializing Image Processing (MIPI CSI-2 Input) The description in the tenth box from the top, modified ([YUV? RGB conversion to be performed?] → [YUV → RGB conversion to be performed?]); The description at the right side of the tenth box from the top, modified ([N (RGB? YUV conversion)] → [N (RGB → YUV conversion)]); Note, modified
		3086	Figure 35.38 Initializing the LINK (MIPI CSI-2 Input) The descriptions in the second box and the last box from the bottom, modified
		3089	Figure 35.41 Stopping the D-PHY (MIPI CSI-2 Input) DPHYCTRL0 → CSIDPHYCTRL0
		3090	Figure 35.42 Stopping Image Processing Reception (MIPI CSI-2 Input) AMnAXISTPL → AMnAXISTP; FIFOPOINTER_Y → FIFORPNTR_Y; FIFOPNTR → FIFOWPNTR; FIFOPOINTER_UV → FIFORPNTR_UV; FIFWPNTR[7:1] → FIFOWPNTR[7:1]
		3093	Figure 35.44 Flow for Changing the Frame Synchronization Settings (by Using the DMAC) The description in the third box from the top, modified

Rev.	Date	Description Page	 Summary
1.10	Apr 26, 2022	3096	Figure 35.47 Initializing Image Processing (Parallel Input) The description in the ninth box from the top, modified ([YUV? RGB conversion to be performed?] → [YUV → RGB conversion to be performed?]); The description at the right side of the ninth box from the top, modified ([N (RGB? YUV conversion)] → [N (RGB → YUV conversion)])
		3099	Figure 35.50 Stopping Image Processing Reception (Parallel Input) AMnAXISTPL → AMnAXISTP; FIFOPOINTER_Y → FIFORPNTR_Y; FIFOPNTR → FIFOWPNTR; FIFOPOINTER_UV → FIFORPNTR_UV; FIPOPNTR → FIPOWPNTR; FIFWPNTR[7:1] → FIFOWPNTR[7:1]
		3100	Figure 35.51 Flow for Stopping the Pattern Generator AMnAXISTPL → AMnAXISTP
		36. Image Scaling Unit (ISU)	
		3133	Figure 36.39 Normalization for YCbCr/YUV420 Format, modified
		3137	36.2 Register Configuration: The description, modified
		3137	36.2.1 System Management Register The Address column in the table, modified
		3141	36.3.3 FM Frame Control Registers (ISU_FM_FRCON) The table of bits: Description of the bit 0 (START), modified
		3142	36.3.4 FM Module Stop Registers (ISU_FM_STOP) The table of bits: Description of the bit 0 (STOP), modified
		3156	36.3.19 RPF Source TEST Data Registers 1 (ISU_RPF_SRC_TD1) The bit chart: Bit Name of the bit 18, modified (GRADA_B → GRADA_R); Bit Name of the bit 16, modified (GRADA_R → GRADA_B)
		3157	36.3.20 RPF Source TEST Data Registers 2 (ISU_RPF_SRC_TD2) The bit chart: Bit Name of the bits 23 to 16, modified (INIT_B[7:0] → INIT_R[7:0]); Bit Name of the bits 7 to 0, modified (INIT_R[7:0] → INIT_B [7:0])
		3170	36.3.36 WPF Color Collection MUL Coefficient Registers2 (ISU_WPF_MUL2) The bit chart: Bit Name of the bits 29 to 16, modified (K13 → K12); Bit Name of the bits 13 to 0, modified (K12 → K13) The table of bits: Bit Name of the bits 29 to 16, modified (K13 → K12); Bit Name of the bits 13 to 0, modified (K12 → K13) The table of bits: Description of the bits 29 to 16, modified (K13 → K12); Description of the bits 13 to 0, modified (K12 → K13)
		3171	36.3.38 WPF Color Collection MUL Coefficient Registers4 (ISU_WPF_MUL4) The bit chart: Bit Name of the bits 29 to 16, modified (K23 → K22); Bit Name of the bits 13 to 0, modified (K22 → K23) The table of bits: Bit Name of the bits 29 to 16, modified (K23 → K22); Bit Name of the bits 13 to 0, modified (K22 → K23) The table of bits: Description of the bits 29 to 16, modified (K23 → K22); Description of the bits 13 to 0, modified (K22 → K23)
		3172	36.3.40 WPF Color Collection MUL Coefficient Registers6 (ISU_WPF_MUL6) The bit chart: Bit Name of the bits 29 to 16, modified (K33 → K32); Bit Name of the bits 13 to 0, modified (K32 → K33) The table of bits: Bit Name of the bits 29 to 16, modified (K33 → K32); Bit Name of the bits 13 to 0, modified (K32 → K33) The table of bits: Description of the bits 29 to 16, modified (K33 → K32); Description of the bits 13 to 0, modified (K32 → K33)
		3183, 3184	36.4.3 Setting the vertical size, start position and reduction factor, added
		xxxx	Figure 36.45 Schematic Diagram of IP Conversion, modified
		3184	Table 36.12 Image Size Conversion Example 1 Description of Processing method, modified
		3196	Figure 36.52 Image Size Conversion Example 4 0x5020_0000 → H’5000_0000

Rev.	Date	Description	
		Page	Summary
1.10	Apr 26, 2022	40. SD/MMC Host Interface	
		3213	40.1.1 Features DDR50, deleted from the supported transfer modes; SD clock (SD_CLK) frequency, modified
		3234	40.2.10 SD Clock Control Register (SD_CLK_CTRL) The table of bits: Description of the bits 7 to 0, modified (P1ϕ → (SDxϕ /4)); (x = 0,1), added
		3297	40.6.3 SCC Register Configuration: The description, modified
		41. General Purpose Input Output Port (GPIO)	
		3309	41.1.3 Special Purpose Port Function Control Target Signals, modified (TDO/SWO, WDTOVF_PERROUT → TDO, WDTOVF_PERROUT#)
		3310	Figure 41.1 P0_0 to P19_1, P38_0 to P48_4 (Multiplexed peripheral functions), modified
		3315	41.1.5 Special purpose port Configuration Target Signals, modified (TDO/SWO, WDTOVF_PERROUT → TDO, WDTOVF_PERROUT#)
		3320	The title of the figure, modified (41.12 (12) TDO/SWO, WDTOVF_PERROUT → 41.12 (12) TDO, WDTOVF_PERROUT#)
		3352	41.3.7 Driving Ability Control Register (IOLH) The table of bits: Description of the bits 25, 24, 17, 16, 9, 8, 1, and 0, modified ((IOLH7 to IOHL0) → (IOLH4 to IOLH0))
		3367	41.3.14 Ether ch0 Voltage Mode Control Register (ETH_ch0) The description, modified ((ETHER0_P VDD) → (PVDD182533_0))
		3368	41.3.15 Ether ch1 Voltage Mode Control Register (ETH_ch1) The description, modified ((ETHER1_P VDD) → (PVDD182533_1))
		3369	41.3.16 Ether MII/RGMII Mode Control Register (ETH MII/RGMII) The bit chart: Bit Name of the bit 1, modified (P29_0 → ETH1_mode); Bit Name of the bit 0, modified (P20_0 → ETH0_mode) The table of bits: Bit Name of the bit 1, modified (P29_0 → ETH1_mode); Bit Name of the bit 0, modified (P20_0 → ETH0_mode)
		42. Low Power Mode	
		3380	Table 42.3 Registers for Module Standby Mode (2/2) Target module for Module Standby Mode: USB2.0 ch0/ch1, deleted; USB2.0 PHY, added; Note 2, added
		43. Debug Interface	
		3395	Table 43.6 Handling of Unused Pins: Note 2, added
		44. Thermal Sensor Unit (TSU)	
		3406	Figure 44.1 Block Diagram of the TSU, modified
		3409	44.2 Register Configuration The description, modified (Cortex-A33 → Cortex-M33)
		3414	44.3.5 TSU OTP Calibration Register 0 (OTPTSUTRIM0_REG) The bit chart: Bit Name of the bit 31, modified (OTPTSUTRIM0 → OTPTSUTRIM0_EN); Bit Name of the bits 30 to 12, modified (OTPTSUTRIM0 → —) The table of bits: Bit Name of the bit 31, modified (OTPTSUTRIM0 → OTPTSUTRIM0_EN); Bit Name of the bits 30 to 12, modified (OTPTSUTRIM0 → —) The table of bits: Description of the bits 31 to 12, modified
		3415	44.3.6 TSU OTP Calibration Register 1 (OTPTSUTRIM1_REG) The bit chart: Bit Name of the bit 31, modified (OTPTSUTRIM1 → OTPTSUTRIM1_EN); Bit Name of the bits 30 to 12, modified (OTPTSUTRIM1 → —) The table of bits: Bit Name of the bit 31, modified (OTPTSUTRIM1 → OTPTSUTRIM1_EN); Bit Name of the bits 30 to 12, modified (OTPTSUTRIM1 → —) The table of bits: Description of the bits 31 to 12, modified

Rev.	Date	Description	
		Page	Summary
1.10	Apr 26, 2022	45. Trusted Secure IP	
		3419	Table 45.1 Specifications of Trusted Secure IP (1/2) Description of Encryption engine, modified (ECC, added)
		3421	Figure 45.1 Trusted Secure IP Block Diagram: ECC, added
		48. Electrical Characteristics	
		3436	Table 48.3 DC Characteristics (1) [3.3-V I/O] Weak pull-up resistor (input mode) and Weak pull-down resistor (input mode), added
		3436	Table 48.4 DC Characteristics (2) [1.8-V I/O] PV_{DD18} → V_{DD18}
		3436	Table 48.6 DC Characteristics (5) [1.8-V Input] PV_{DD18} → V_{DD18}
		3437	The title of the table, modified (48.7 DC Characteristics (6) [RGMII] → 48.7 DC Characteristics (6) [RGMII/MII]); VDV_{DD} → $PV_{DD182533\_n}$; VDV_{SS} → V_{SS}; RGMII → RGMII/MII
		3442	48.5 AC Characteristics: Conditions, modified
		3442	Table 48.20 Clock Timing Table EXTAL → EXCLK
		3442	The title of the figure, modified (48.4 EXTAL, AUDIO_CLK1 and AUDIO_CLK2 Clock Input Timing → 48.4 EXCLK, AUDIO_CLK1 and AUDIO_CLK2 Clock Input Timing); EXTAL → EXCLK
		3443	Figure 48.5 Power-On Oscillation Settling Time TRST#, deleted
		3453	Table 48.30 Ethernet-IF Access Timing (Ether MII) Note, modified
		3454	Table 48.31 Ethernet-IF Access Timing (Ether RGMII) Rise/fall time → Rise/fall time (20-80%); The units of Duty_G and Duty_T, modified (ns → %); Note 2, added
		3461	The order to list the figures, modified (Figure 48.33 Reset Input Timing 2 → Figure 48.32 Reset Input Timing 2; Figure 48.32 Interrupt Signal Input Timing → Figure 48.33 Interrupt Signal Input Timing)
		3463	Table 48.35 SSIF-2 Timing Note, added
		3465	Table 48.36 CAN-FD Interface Timing Note → Note 1; Note, added
		3466	Table 48.37 MTU3a Timing Note → Note 1; Note, added
		3467	Table 48.39 GPT Timing Note, added
		3469	Table 48.41 I2C Bus Interface Timing Note, added; Note 5, modified
		3471	Table 48.42 SCIFA Timing Note, added; Unit, modified (t_{SEcyc} → t_{p1cyc})
		3473	Table 48.43 SCI Timing Note, added; Unit, modified (t_{SEcyc} → t_{p1cyc})
		3475	Table 48.44 Renesas Serial Peripheral Interface Timing Note, added
1.20	Sep 26, 2022	4. Boot Mode	
		105	4.1 Overview: Note added for starting the user program

Rev.	Date	Description	
		Page	Summary
1.20	Sep 26, 2022	7. Clock Pulse Generator (CPG)	
		515	7.4.6 Procedure for Switching the Division Ratio of the Dynamic Switching Frequency Dividers: The description “Division ratio setting (PLL1) register (CPG_PL1_DDIV)”, added
		34. MIPI DSI	
		2919	34.3.2.2 DSI D-PHY Timing Register 0 (DSIDPHYTIM0): The description on bits 18 to 0 (T_INIT), modified. Note 1, added.
		2919	Table 34.6 Recommended Setting Values of the DSI D-PHY Timing Register, added
		2920	34.3.2.3 DSI D-PHY Timing Register 1 (DSIDPHYTIM1): The description on bits 31 to 24 (THS_PREPARE) and bits 23 to 16 (TCLK_PREPARE), modified. Note 1, added.
		2921	34.3.2.4 DSI D-PHY Timing Register 2 (DSIDPHYTIM2): The description on bits 31 to 24 (TCLK_TRAIL), bits 23 to 16 (TCLK_POST), bits 15 to 8 (TCLK_PRE), and bits 7 to 0 (TCLK_ZERO), modified. Note 1, added.
		2922	34.3.2.5 DSI D-PHY Timing Register 3 (DSIDPHYTIM3): The description on bits 31 to 24 (TLPX), bits 23 to 16 (THS_EXIT), bits 15 to 8 (THS_TRAIL), and bits 7 to 0 (THS_ZERO), modified. Note 1, added.
		35. Camera Data Receiving Unit (CRU)	
		2962	Table 35.2 Supported Image Format and Maximum Transfer Rate: No.12, ICnMC.ICTHR (the column), deleted
		2967	35.1.6.2 Module Control Register 0 (CSI2nMCT0): The R/W attribute and description on bit 20, modified
		2976	35.1.6.12 Receive Status Clear Register (CSI2nRXSC): The R/W attribute on bit 24, modified
		2977	35.1.6.13 Receive Interrupt Enable Register (CSI2nRXIE): The R/W attribute on bit 24, modified
		2980	35.1.6.15 Data Lane (N) Status Clear Register (CSI2nDLSC(N)): The R/W attribute on bits 19, 18, 11, 10, 9, and 8, modified
		2981	35.1.6.16 Data Lane (N) Interrupt Enable Register (CSI2nDLIE(N)): The R/W attribute and description on bits 19, 18, 11, 10, 9, and 8, modified
		2993	35.1.6.23 Generic Short Packet Control Register (CSI2nGSCT): The R/W attribute and description on bit 17, modified
		2998	35.1.6.29 D-PHY Control Register 0 (CSIDPHYCTRL0): The R/W attribute on bit 16, 8, and 2, modified
		2999	Table 35.7 Recommended Setting Values of the D-PHY Timing Register: The transmission rate for No. 3 to 6, modified
		3010, 3011	35.2.1.1 Features: The description on the operating ranges, output data format (type), image format conversion, and conversion by using a lookup table (LUT), modified
		3012	Figure 35.8 Block Diagram of the Image Processing Module, modified
		3013	35.2.1.3, (1) Parallel Data Input Timing: The description of the example settings in Figure 35.9 and Figure 35.10, added
		3019	35.2.3.1 CRU Control Register (CRUnCTRL): The R/W attribute and description on bit 1, modified
		3020	35.2.3.2 CRU Interrupt Enable Register (CRUnIE): The R/W attribute and description on bits 31 to 29, and 11 to 8, modified
		3022	35.2.3.3 CRU Interrupt Status Register (CRUnINTS): The R/W attribute and description on bits 31 to 29, and 11 to 8, modified
		3034	35.2.3.24 Memory Bank Enable Register for CRU Image Data (AMnMBVALID): The description on bits 7 to 0 (MBVALID[7:0]), modified
		3035	35.2.3.26 AXI Master Transfer Setting Register for CRU Image Data (AMnAXIATTR): The R/W attribute and description on bits 27 to 24, 22 to 16, 13 to 12, 9 to 8, and 6 to 4, modified. The description on bits 3 to 0 (AXILEN[3:0]), modified.

		Description	
Rev.	Date	Page	Summary
1.20	Sep 26, 2022	3043 to 3045	35.2.3.34 CRU Image Processing Main Control Register (ICnMC): The R/W attribute and description on bits 25 to 24, and 15, modified. The description on bits 13 to 12 (CLP[1:0]), modified. The R/W attribute and description on bit 7, modified. The description and note on bit 6 (LUTTHR), modified. The note on bit 5 (CSCTHR), modified. The R/W attribute and description on bits 4 to 3, modified. The note on bit 2 (CLPTHR), modified. The note on bit 1 (DECTHR), modified. The bit name (ICTHR → Reserved) and description on bit 0, modified.
		3045	Figure 35.11 Bypass Setting of Each Function, modified
		3050	35.2.3.40 CRU Parallel I/F Control Register (ICnPIFC): The description on bit 14 (ENPOL) and bit 13 (VSPOL), modified. Note 1 was modified and Note 2 and Note 3 were added for bits 10 to 8 (ITL[2:0]).
		3062	35.2.3.54 CRU RGB → YCbCr Color Space Conversion Y Coefficient 3 Register (ICnYCCR3): The R/W attribute and description on bits 31 and 16, modified
		3064	35.2.3.57 CRU RGB → YCbCr Color Space Conversion Cb Coefficient 3 Register (ICnCBCCR3): The R/W attribute and description on bits 31 and 16, modified
		3066	35.2.3.60 CRU RGB → YCbCr Color Space Conversion Cr Coefficient 3 Register (ICnCRCCR3): The R/W attribute and description on bits 31 and 16, modified
		3074	Table 35.12 Input Format and Image Processing (1/2), modified: All Bypassed (ICnMC.ICTHR = 1) (the column), deleted. MIPI CSI-2 Recommended Memory Storage Format (the column), added. Parallel User-defined binary 16-bit (progressive) (the column), added
		3075, 3076	Table 35.12 Input Format and Image Processing (2/2), modified: All Bypassed (ICnMC.ICTHR = 1) (the column), deleted. MIPI CSI-2 Recommended Memory Storage Format (the column), added. Parallel User-defined binary 16-bit (progressive) (the column), added. Note 2 and Note 3, added.
		3077	Table 35.13 Image Output Formats (1/2), modified: MIPI CSI-2 Recommended Memory Storage Format, Parallel User-defined binary 16-bit (progressive), added
		3078	Table 35.13 Image Output Formats (2/2), modified: MIPI CSI-2 Recommended Memory Storage Format, Parallel User-defined binary 16-bit (progressive), added
		3083	35.2.4.3, (3) Image Data Transfer Destination Address (Storage in Interlaced Mode), (d) Memory bank write order by CAM_FIELD pin input and ICnPIFC.ITL[2] register setting: Added
		3085	35.2.4.5 Parallel User-defined binary 16-bit (progressive), added
		3085	35.2.4.6 RGB565/RGB666 to RGB888, added
		3085	35.2.4.7 MIPI CSI-2 Recommended Storage format, added
		3088	35.2.4.11 Data Clipping: The description for YCbCr-10bit, added
		3102	Figure 35.36 Initializing Image Processing (MIPI CSI-2 Input): “ICnMC.ICTHR” in “Set modes”, deleted
		3113	Figure 35.47 Initializing Image Processing (Parallel Input): “ICnMC.ICTHR” in “Set modes”, deleted
1.30	May 12, 2023	All	Hexadecimal notation, unified to H'xx
		1. Overview	
		71	1.2.2 CPU Peripheral: Interrupt Controller: Note 1, added
		78	1.2.11 Timer: General PWM Timer (GPT): The number of channels for [RZ/G2LC], changed
		79	1.2.12 Peripheral Module: Renesas Serial Peripheral Interface (RSPI): The description on the maximum transfer rate, deleted
		2. System CPU Cortex-A55	
		83	Table2.2 Core Configuration: AArch32, deleted
		3. System CPU Cortex-M33	
		102	3.4.4 Cortex-M33 Sleep Mode: The location of *^{3}, modified
		4. Boot Mode	
		105	4.1 Overview: Note, added

Rev.	Date	Description	
		Page	Summary
1.30	May 12, 2023	6. System Controller (SYSC)	
		173	6.3.3 Master Access Control Register 2 (SYS_MSTACCCTL2): The bit name of bits 19, 11, and 3, modified
		203	6.3.41 LSI Mode Signal Register (SYS_LSI_MODE): Bit 16, changed (Reserved → the STAT_SEC_EN bit)
		204	6.3.42 LSI Device ID Register (SYS_LSI_DEVID): The section title, modified
		7. Clock Pulse Generator (CPG)	
		242	Table 7.5 Register list (4/5): Monitor Register SYC (CPG_RSTMON_SYC): The initial value and the address, modified
		—	Table 7.5 Register list (4/5): Enable Pin Control Register GTM (CPG_EN_GTM), deleted
		243	Table 7.5 Register list (5/5): Clock Control Register MBUS (CPG_CLKON_MBUS) and Clock Monitor Register MBUS (CPG_CLKMON_MBUS), added
		250	7.2.4.3 PLLn (SSCG) Output Clock Setting Register 2 (CPG_SAMPLLn_CLK2): Note 3, added
		257	7.2.4.9 PLL5 (SSCG) Standby Control Register (CPG_SIPLL5_STBY): The description of the register, modified; Note on enabling PLL5, added
		259	7.2.4.10 PLL5 (SSCG) Output Clock Setting Register 1 (CPG_SIPLL5_CLK1): Bit 24 (REFDIV_WEN), bit 20 (POSTDIV2_WEN), and bit 16 (POSTDIV1_WEN), changed to reserved bits
		265	7.2.4.14 PLL5 (SSCG) Output Clock Setting Register 5 (CPG_SIPLL5_CLK5): The calculation formulae for FOUTVCO and FOUTPOSTDIV, added
		272	7.2.4.19 Division Ratio Setting (PLL3B) Register (CPG_PL3B_DDIV): The description of bits 2 to 0 (DIVPL3CLK200FIX_SET): Note, added
		310	7.2.4.51 Clock Control Register CRU (CPG_CLKON_CRU): Bit 20 (CLK4_ONWEN) and bit 4 (CLK4_ON), changed to reserved bits
		355	7.2.4.91 Clock Monitor Register CRU (CPG_CLKMON_CRU): Bit 20(CLK4_ONWEN) and bit 4(CLK4_ON), changed to reserved bits
		—	7.2.4.187 Enable Pin Control Register GTM (CPG_EN_GTM), deleted
		457	7.2.4.187 WDT Reset Selector Register (CPG_WDTRST_SEL): Bit 26 (WDTRSTSEL10_WEN), bit 25 (WDTRSTSEL9_WEN), bit 24 (WDTRSTSEL8_WEN), bit 22 (WDTRSTSEL6_WEN), bit 21 (WDTRSTSEL5_WEN), bit 20 (WDTRSTSEL4_WEN), bit 18 (WDTRSTSEL2_WEN), bit 17 (WDTRSTSEL1_WEN), and bit 16 (WDTRSTSEL0_WEN), changed to reserved bits
		488	7.2.4.207 Clock Control Register MBUS (CPG_CLKON_MBUS), added
		489	7.2.4.208 Clock Monitor Register MBUS (CPG_CLKMON_MBUS), added
		494	Table 7.12 Range of Reset Application (1/2): No. 13: WDT (Reset for WDTOVF Register), changed (Reset → Released)
		518	7.4.7 Procedure for Switching Clocks by the Dynamic Switching Frequency Selectors: 2) Setting the switching of the clock output from SEL_PLL4: Setting of clocks switching: The description, modified
		519	7.4.8 Procedure for Switching Clocks by the Static Switching Frequency Dividers and Selectors (Procedure for Switching Source Clocks): Note 1 and Note 2, added
		8. Interrupt Controller	
		557, 558	8.6.18 ECCRAM Error Interrupt Status Control Register0 (EREISR0): Bit in the table of bits, modified
		559, 560	8.6.19 ECCRAM Error Interrupt Status Control Register1 (EREISR1): Bit in the table of bits, modified
		10. On-chip RAM	
		573	The description (bit), modified
		16. Multi-Function Timer Pulse Unit 3 (MTU3a)	
		901	Figure 16.82 Example of Output Phase Switching through UF, VF, and WF Bit Settings (2): The setting values, modified

Rev.	Date	Description: Page	Description: Summary
1.30	May 12, 2023	19. Port Output Enable for GPT (POEG)	
		1205	19.2.1 POEG Group n Setting Register (POEGGn) (n = A to D): Bit chart: Bit16: The bit name, modified
		1209	19.3.2 Output-Disable Requests from the GPT: The reference, modified
		21. Watchdog Timer (WDT)	
		1255	Table 21.6 WDT Reset Target by CPG_WDTRST_SEL Register: The lines for Reserved, deleted
		22. Serial Communications Interface with FIFO (SCIFA)	
		1318	Figure 22.24 Receive Data Sampling Timing in Asynchronous Mode: Receive data (RXD), modified
		23. Serial Communications Interface (SCIg)	
		1321	Figure 23.1 Block Diagram of SCIg (SCI0 and SCI1): The pin name, modified (RTSn#/ → RTSn#, CTSn#/ → CTSn#)
		1390	Figure 23.33 Receive Data Sampling Timing in Smart Card Interface Mode (When Clock Frequency is 372 Times the Bit Rate): Receive data (RXDn), modified
		24. Renesas Serial Peripheral Interface	
		1425	Table 24.3 Relationship between SPBR and BRDV1 and BRDV0 Settings: The location of *^{1}, modified. The bit rate for SPBR(n) = 0, modified.
		1435	Table 24.4 Relationship between Modes and SPCR and Description of Each Mode: Master (SPI Operation) for Transfer rate (in Mode), modified
		25. SPI Multi I/O Bus Controller	
		1537	Figure 25.29 (2) Flow of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode): The figure title, modified. 13-step timing setting → 8-step timing setting. The values for y zz x, modified
		1538	Figure 25.30 (3) Example of Timing Adjustment when the Serial Flash Memory is Connected (DDR Mode): The title, modified
		26. I^{2}C Bus Interface	
		1550	26.3.1 RIICnCR1 − I^{2}C Bus Control Register 1: The description of bit 3 (SCLO), modified (SDA Output Control → SCL Output Control)
		1603	Figure 26.15 Example of Slave Transmission Flowchart: The judgement, modified (ansmitted? → All data transmitted?)
		30. Gigabit Ethernet Interface	
		2122	30.4.1.21 Error Interrupt Disable Register (EID): The description (register name), modified
		32. USB2.0	
		2377	Table 32.6 USBPHY Control Register List: VBUSEN control register, added
		2381	32.3.4 VBUSEN control register (VBENCTL), added
		32A. USB 2.0 Host Module	
		2392	32A.1.2.3 OTG function: The description of the OTG_PERI bit, modified
		2439	32A.2.4.3, (3) USBINTR Register: The section title, modified
		2464	32A.2.4.5, (1) Common Control Register: The description of bit 31 (OTG_PERI), modified
		2465	32A.2.4.5, (2) OTG-BC Interrupt Status Register: Bit 3 (VBSTAINT_STA) and bit 2 (VBSTACHG_STA), changed to reserved bits
		2466	32A.2.4.5, (3) OTG-BC Interrupt Enable Register: Bit 3 (VBSTAINT_EN) and bit 2 (VBSTACHG_EN), changed to reserved bits
		2467	32A.2.4.5, (4) VBUS Control Register: Bit 0: changed (Reserved → the VBOUT bit)
		2468	32A.2.4.5, (5) Line Control Port 1 Register: Bits 25 and 24 (VDPMLLVL[1:0]), changed to reserved bits
		2476	32A.4.2.3 U2H_EHCI_INT assertion source and control: [Interrupt enable control by register]: The description (bit), modified. [Interrupt source]: The row of bit [17], deleted. The description of bit[16] and bit[2], modified

Rev.	Date	Description	
		Page	Summary
1.30	May 12, 2023	2478	32A.4.2.5 U2H_OBINT assertion source and control: [Interrupt enable control by register]: The description (bits), modified; [Interrupt source]: The rows of bit [3], bit [2], deleted; The description of the VBUSIN pin for Interrupt Source (1, 2), deleted; [Clearing the interrupt]: The description (bits), modified
		2486	32A.8.3 Port Power (VBUS) control specifications: The description, modified
		33. LCDC	
		2675	Table 33.1 Sub modules in FCPVD: The description of COMP, modified
		2687	Table 33.6 FCPVD Register Configuration: FCPVD configuration register 0, FCP reset register, and FCP status register, deleted
		2691	Table 33.8 DU Register Configuration: DU Display I/F Timing Register 5, deleted
		—	33.3.1.2 FCPVD Configuration Register 0 (FCP_CFG0), deleted
		—	33.3.1.3 FCP Reset Register (FCP_RST), deleted
		—	33.3.1.4 FCP Status Register (FCP_STA), deleted
		2702	33.3.2.4, (7) Software Reset Register (VI6_SRESET): Note 5, deleted
		2725	Figure 33.7 Chrominance Interpolation Methods Selectable through CIPM Setting: VSP → VSPD
		2789	33.3.2.10, (4) Color Information Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_COL): The description (unit), modified
		2805	33.3.3.8 DU Display I/F Timing Register 3 (DU_DITR3): The description of bits 11 to 0 (HSA), modified
		2806	33.3.3.9 DU Display I/F Timing Register 4 (DU_DITR4): The description of bits 28 to 16 (HFP), and bits 12 to 0 (HBP), modified
		—	33.3.3.10 DU Display I/F Timing Register 5 (DU_DITR5), deleted
		2807	33.3.3.10 DU Module Control Register 1 (DU_MCR1), Bit 8 (UPD_SZ), changed to a reserved bit
		—	33.4.1(a) FCPVD initialization, deleted
		—	33.4.1(c) To stop image process immediately, deleted
		2811	33.4.2 Reset Operation: The description, modified
		—	33.4.2.1 Software Reset, deleted
		2811	33.4.3 Registers to set fixed value: The description of [11], added
		2816	Table 33.31 Definition of a Function for Simple Description: Syntax; I = 0; i<num_bits; i++) → i = 0; i<num_bits; i++)
		2825	33.4.8.1 Operation flow of VSPD and DU: The description, modified (VSP2 → VSPD)
		2826	Figure 33.40 Updating display list at arbitrary timing: VSP2 → VSPD
		2826	Figure 33.41 Controlling Two Register Planes Using a Display List: The description, added; VSP1 → VSPD, VSP → VSPD
		2827	Figure 33.42 Input/output Size for Each Module in a Sample DPR Connection: VSP → VSPD
		—	Figure 33.41 Setting flow to stop VSPD linked with DU, deleted
		34. MIPI DSI	
		2844	34.3.1.3 Tx Set Register (TXSETR): The description of bits 17 and 16 (NUMLANECAP), modified
		35. Camera Data Receiving Unit (CRU)	
		2961	Table 35.4 Register Configuration of MIPI CSI-2 LINK: The table header, modified (Address → Offset Address)
		3066	35.2.3.63 CRU Lookup Table Pointer Register (ICnLUTP): Bits 29 to 20: Bit Name: LTYPR[9:0] → LTCRPR[9:0]; The description, modified. Bits 9 to 0: Bit Name: LTCRPR[9:0] → LTYPR[9:0]; The description, modified
		3067	35.2.3.64 CRU Lookup Table Data Register (ICnLUTD): Bits 23 to 16: Bit Name: LTYDT[7:0] → LTCRDT[7:0]; The description, modified. Bits 7 to 0: Bit Name: LTCRDT[7:0] → LTYDT[7:0]; The description, modified

Rev.	Date	Description	
		Page	Summary
1.30	May 12, 2023	3085	35.2.4.10 Data Arrange: The description, modified
		3086	35.2.4.12, (1) RGB to YCbCr: Register Settings: The register settings, modified
		3092	35.2.4.16, (5) Generated Image Data Example (in the Case of Pattern Generation): The description (the setting values for TIPTNV1[3:0]), modified
		36. Image Scaling Unit (ISU)	
			All: collection → correction
		3118	Table 36.2 Descriptor Format: Item: Setting Range of Register Address 2, 3, 4, modified
		3124	Table 36.5 Output Image Setting: The register of No. (1), (2), modified. The variation for No. (4), modified
		3126	36.1.3.2 Formatter: The section title, modified (Data Extraction → Formatter). The description, modified
		3129	36.1.3.3 Data Extraction: The section title, modified (Formatter → Data Extraction). The description, modified
		3141	Figure 36.32 Color Correction Equation, modified
		3149	Figure 36.43 Data Swap Correction: The byte address, modified (RD_SWAP_*** → WD_SWAP_***)
		3161	36.3.8 AXI FIFO Capability Registers (ISU_AXI_FIFO_CAP): The initial values of bits 30 to 24 (SCAP_PL0[6:0]), and bits 14 to 8 (DCAP_PL0[6:0]), modified
		3166	36.3.15 RPF Source Image Format Registers (ISU_RPF_FMT): The reference, modified; Bit 5 to 0 in the table of bits, modified
		3167	36.3.16 RPF Source Image UV Format Registers (ISU_RPF_UVBIN): The reference, modified
		3168	36.3.18 RPF Source ALPHA Data Selection Registers (ISU_RPF_ALPH_SEL): The reference, modified
		3181	36.3.33 WPF Destination Image Format Registers (ISU_WPF_FMT): The reference, modified
		3193	36.3.49 AXI Max Burst Length Registers (ISU_AXI_BLEN): Bit 19 to 16(ARLEN_MAX[3:0]): The initial value of bits 3 to 0 (AWLEN_MAX[3:0]), modified
		3194	Table 36.11 Explanation for ISU FM Frame Control Registers (ISU_FM_FRCON): The table title, modified
		3194	Figure 36.44 Relationship Between Register ISU_FM_FRCON Setting and Frame Processing Mode: The figure title, modified
		3199	Table 36.12 Image Size Conversion Example 1: The description of Input format and Output format, modified
		3204	Table 36.16 Image Size Conversion Example 2: The description of Input format, Output format, and Output stride, modified
		3208	Table 36.20 Image Size Conversion Example 3: The description of Input format, Output format, and Output stride, modified
		3212	Table 36.25 Image Size Conversion Example 4 (Register Settings): The comment of ISU_RPF_FMT, modified
		3215	36.5.3.2 YCbCr422 → ARGB8888: The conversion formula from YCbCr422 to RGB, modified
		3217	36.5.3.3 BGR888 → YCbCr422: The section title, modified
		3217	Table 36.30 Color Format Conversion Example 3: The description of Input stride, modified
		40. SD/MMC Host Interface	
		3228	40.1.1 Features: The channels, added. NOTE, deleted.
		3234	40.2.2 Command Argument Register (SD_ARG): SD_ARG: Bits 63 to 32 (Reserved), added. SD_ARG1: Bits 63 to 16 (Reserved), added
		3238	40.2.5 SD Card Response Registers (SD_RSP): SD_RSP5, modified (31 to 16 (Reserved) → 63 to 16 (Reserved))
		3249	40.2.10 SD Clock Control Register (SD_CLK_CTRL): The description of bits 7 to 0 (DIV7 to DIV0), modified

		Description	
Rev.	Date	Page	Summary
1.30	May 12, 2023	3252	40.2.12 SD Card Access Control Option Register (SD_OPTION): The description of bit 15 (WIDTH), modified
		3261	40.2.23 SD Interface Mode Setting Register (SDIF_MODE): The description (mode), modified. Bit 0 (DDR), changed to a reserved bit
		—	Figure 40.4 SDDAT in 4-Bit Width DDR Mode, deleted
		—	Figure 40.6 SDDAT in 8-Bit Width DDR Mode, deleted
		3271	40.3.1, (2) Bus Signal Voltage Switch: The description of (4), modified
		3273	Figure 40.7 SD Card Detect with SDDAT3: The card detect, modified (2P1ϕ → SDxϕ/2 (x = 0, 1))
		3307	Table 40.9 Example of SD_CMD Register Setting (MMC): The remark of CMD21, deleted
		3312	40.6.3 SCC Register Configuration: Base Address of Cortex-M33, added
		42. Low Power Mode	
		3395	Table 42.3 Registers for Module Standby Mode: The register name, modified (OSTM → GTM)
		45. Trusted Secure IP	
		3436	Table 45.1 Specifications of Trusted Secure IP: RSA for Encryption engine: Number of cycles for execution, added
		48. Electrical Characteristics	
		3450	Figure 48.1 Power-On/Power-Off Sequence, modified (PVDD182533 → PVDD182533_0/1, PLL1/2/3/4/5/6_AVDD18 → PLL1/23/4/5/6_AVDD18)
		3456	Table 48.18 DC Characteristics (16) [ADC]: The Max. value of Analog input channel, changed
		3459	Figure 48.5 Power-On Oscillation Settling Time: CPUHOLD、SEC_EN, deleted
		3460	48.5.2.1 SDHI Access Timing (SDR 3.3-V): NOTE, modified
		3460	Table 48.21 SDHC AC Access Timing (SDR at 3.3-V Operation): The notation of Symbol, modified (T*** → t***)
		3476	Figure 48.28 Timing for Switching the Buffers On and Off, modified (t_{SOFF} → t_{BOFF})
		3490	48.5.19 Serial Communications Interface (SCIg) Access Timing: The section title, modified (SCI → SCIg)
		3490	Table 48.44 SCIg Timing: The table title, modified (SCI → SCIg)
		3492	Table 48.45 Renesas Serial Peripheral Interface Timing: The max. value of Data output delay time (t_{OD}), modified
		3495	Table 48.46 A/D Converter Timing: The unit of ADC Trigger Input Pulse Width, modified
1.40	Sep 29, 2023	1. Overview	
		75	1.2.7 Camera Interfaces: [RZ/G2L] Support Image Processing, modified
		75	1.2.7 Camera Interfaces: [RZ/G2LC] Support Image Processing, modified
		3. System CPU Cortex-M33	
		100	Table 3.3 List of Functions Controlled by the SYSC: The description, modified (Lowpower → Low-power)
		5. LSI Internal Bus	
		154, 155	Table 5.3 Accessible Areas: “CRU [Video]” under the top header “Master Unit” modified to “CRU [Video, Statistics]”
		155	Table 5.3 Accessible Areas (2/2): Slave Unit, changed (TZC (SRAM CPU) → TZC (SRAM ACPU)); TZC (SRAM MCPU), added
		6. System Controller (SYSC)	
		178	6.3.5 Master Access Control Register 4 (SYS_MSTACCCTL4): Bit 27, changed (Reserved → CRU_ST_AWSEL); Bit 25, changed (Reserved → CRU_ST_AWNS); Bit 24, changed (Reserved → CRU_ST_AWPU). The R/W attribute and description on bits 27, 25 and 24, changed
		193	6.3.23 WDT0 Control Register (SYS_WDT0_CTRL): The bit chart: R/W of bit 16, modified

Rev.	Date	Description	
		Page	Summary
1.40	Sep 29, 2023	194	6.3.24 WDT1 Control Register (SYS_WDT1_CTRL): The bit chart: R/W of bit 16, modified
		195	6.3.25 WDT2 Control Register (SYS_WDT2_CTRL): The bit chart: R/W of bit 16, modified
		222	6.3.57 SYS_LP_CA55CK_CTL1: The table of bits: The initial value of bit 0, modified (— → 0b)
		7. Clock Pulse Generator (CPG)	
		245	Table 7.5 Register list: Note 1, modified
		8. Interrupt Controller	
		566	Table 8.6 BEISR1 Bus Error Mapping: The descriptions of BESTAT32 and BESTAT33, changed
		12. Message Handling Unit (MHU)	
		590	Table 12.4 Register attributes: Entries under the “Description” column, modified
		592	Table 12.5 List of Registers (2/4): The offset address, modified (H’0820 (Reserved) → H’082C (Reserved))
		600	12.4.36 Non secure response transmission interrupt Clear register (CA55 Core1 -> CM33): The description, modified (<n>=0-5 → <n=0-5>)
		609	12.4.84 Secure response transmission interrupt Clear register (CA55 Core1 -> CM33): The description, modified (<n>=0-5 → <n=0-5>)
		610	Figure 12.2 Hardware configuration, modified (Message/Response store Area → Message/Response stored Area); typo corrected (trasmit → transmit)
		16. Multi-Function Timer Pulse Unit 3 (MTU3a)	
		722	Table 16.2 MTU Functions (2/2): Interrupt sources of MTU0, modified (Compare match0E → Compare match 0E, Compare match0F → Compare match 0F)
		809	16.2.35, (2) MTU7.TADCR: Address(es), modified (MTU7, TADCR → MTU7.TADCR)
		925	Figure 16.106 Motor Control Circuit Example, modified
		18. General PWM Timer (GPT)	
		All	Hexadecimal notation unified
		1086	18.2.21 General PWM Timer Cycle Setting Register (GTPR): The bit chart: R/W attributes, added
		1095	Figure 18.3 Example setting for a periodic count operation in up-counting by the count clock: The description of processing [Set count direction], modified
		1188	18.7.3 GTIOC Pin Output Negate Control: The description, modified (selected disable output request → selected output disable request)
		21. Watchdog Timer (WDT)	
		All	Typo corrected (eElapsed → Elapsed)
		22. Serial Communications Interface with FIFO (SCIFA)	
		1320	22.8.6 Receive Data Sampling Timing and Receive Margin in Asynchronous Mode: [Equation 1], modified
		23. Serial Communications Interface (SCIg)	
		1326	23.2.2 Receive Data Register (RDR): The bit chart: R/W attributes, added
		1327	23.2.3 Receive Data Register H, L, HL (RDRH, RDRL, RDRHL): The bit chart: R/W attributes, added
		1328	23.2.4 Transmit Data Register (TDR): The bit chart: R/W attributes, added
		1329	23.2.5 Transmit Data Register H, L, HL (TDRH, TDRL, TDRHL): The bit chart: R/W attributes, added
		1330	23.2.7, (1) Non-Smart Card Interface Mode (SCMR.SMIF = 0): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1332	23.2.7, (2) Smart Card Interface Mode (SCMR.SMIF = 1): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1333	Table 23.4 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits: Note 1, modified

Rev.	Date	Description	
		Page	Summary
1.40	Sep 29, 2023	1334	23.2.8, (1) Non-Smart Card Interface Mode (SCMR.SMIF = 0): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1337	23.2.8, (2) Smart Card Interface Mode (SCMR.SMIF = 1): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1339	23.2.9, (1) Non-Smart Card Interface Mode (SCMR.SMIF = 0): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1342	23.2.9, (2) Smart Card Interface Mode (SCMR.SMIF = 1): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1345	23.2.10 Smart Card Mode Register (SCMR): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1346	Table 23.5 Combinations of the SCMR.BCP2 Bit and SMR.BCP[1:0] Bits: Note 1, modified
		1347	23.2.11 Bit Rate Register (BRR): The bit chart: R/W attributes, added
		1349	Table 23.12 BRR Settings for Various Bit Rates (Clock Synchronous Mode): Note, modified (transmitting/receiving → transmitting or receiving)
		1351	23.2.12 Modulation Duty Register (MDDR): The bit chart: R/W attributes, added
		1352	23.2.13 Serial Extended Mode Register (SEMR): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1354	23.2.14 Noise Filter Setting Register (SNFR): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		1355	23.2.15 Extended function control register (SECR): The bit chart: R/W attributes, added; The table of bits: Initial Value, added
		27. Serial Sound Interface (SSIF-2)	
		1655	Figure 27.8 Alignment Setting of Placement Data in Transmission, modified (010 to 100 → 010 to 101)
		1656	Figure 27.9 Alignment Setting of Placement Data in Reception, modified (010 to 100 → 010 to 101)
		1661, 1664, 1666, 1668	27.4.1.2 Status Register (SSISR): TUIRQ Bit, TOIRQ Bit, RUIRQ Bit, and ROIRQ Bit: [Clearing condition], modified (When either the following → When either of the following)
		1669	Figure 27.18 ROIRQ Setting Timing: The waveform of SSIn_RCK, modified
		1675	Figure 27.22 Timing chart when it begins to communicate master from system reset: The place of the description of “BCK output enable”, modified
		1695	27.4.1.8 Status Control Register (SSISCR): Hexadecimal notation, unified
		1719	Figure 27.57 Procedure to Resume Communication (CPU Procedure): The place of the description of processing “Writing transmit data to FIFO”, modified
		30. Gigabit Ethernet Interface	
		2147	30.4.1.42 Common Interrupt Enable register (CIE): The table of bits: The descriptions of bit 8 and bit 0, modified
		2198	30.4.3.2 Tx Frame Checksum Enable register (CSR1): The table of bits: The initial value of bit 23, modified
		2200	30.4.3.3 Rx Frame Checksum Enable register (CSR2): The table of bits: The initial Value of bit 23, modified
		2240	30.4.3.37 ARPREQ/Neighbor Solicitations Receive Interrupt Status register (CSFR40): The table of bits, modified (31 to 13 → 31 to 3)
		2276	30.5.6.1, (1) Reception Handling: The description, modified
		2286	Figure 30.31 Supported Frame of IPv4, modified (ICMP/TCP/UDPTCP Data → ICMP/TCP/UDP Data)
		2291	Figure 30.36 Received Frame Filtering Flow (Part3), modified (v4_udp_pt_flg → v6_udp_pt_flg)
		2292	Figure 30.37 Received Frame Filtering Flow (Part4), modified (v4_udp_pt_flg → v6_udp_pt_flg)
		2297	Figure 30.41 Overview of Filtering and Interrupt, modified (xxx_flg → yyy_flg)

Rev.	Date	Description	
		Page	Summary
1.40	Sep 29, 2023	2310	Table 30.20 Interrupt of DMAC: The internal trigger of the interrupt flag bit RIS2.RFFF, modified
		2312	30.5.13.1 Set Up Procedure: The description of action “E-MAC.CXR35”, modified (registor → register)
		32. USB2.0	
		2384	32.3.5 Connection control register (CON_CTRL): The table of bits: The description of bit 4, modified
		2387	32.4.2.1 USB/PHY related pins: The sections for reference, modified
		2389	32.4.3.2 Resume from Direct power down mode: Notation unified (us → µs)
		32A. USB 2.0 Host Module	
		All	Offset values deleted from the section titles
		2397	32A.2.3 Register Overview: The initial value of register HcRhDescriptorB, modified
		2400	32A.2.4.1, (2) HcControl Register: The table of bits: The description of bit 9, modified
		2414	32A.2.4.1, (11) HcBulkHeadED Register: The table of bits: The description of bits 31 to 4, modified (Bulk Head ED → BulkHead ED)
		2423	32A.2.4.1, (19) HcRhDescriptorA Register: The table of bits: The description of bits 7 to 0, modified (Down stream → Downstream)
		2424	32A.2.4.1, (20) HcRhDescriptorB Register: The table of bits: The initial value of bit 17, modified (0 → 1)
		2459, 2460	32A.2.4.4, (6) Suspend Control Register: The bit chart: R/W of bit 22, modified. The table of bits: The description of bit 0, modified (Disable status → disabled status)
		2472	32A.2.4.5, (6) BC Control Port 1 Register: The table of bits: R/W of bit 5 (DCPMODE), modified (R → R/W)
		2488	32A.8.3 Port Power (VBUS) control specifications: The section for reference, modified
		32B. USB 2.0 Function Module	
		2501	Table 32B.3 List of Registers (3/3): The register name of the symbol “DSCITVL”, modified (Descriptor Interval → Descriptor Interval Register)
		2512	32B.2.7.3 CFIFO Port Select Register [CFIFOSEL] <Address: H’020>: The table of bits: The initial value of bit 11 and bit 10 (MBW), modified (0 → All 0)
		2574	32B.2.22.1 Peripheral L1 Control Register 1 [PL1CTRL1] <Address: H’144>: The bit chart: R/W of bit 14 (L1EXTMD), added.
		2609	32B.8.7.1 DMA Status SUS Register [DSTAT_SUS] <Address: H’720>: The bit chart: The bit names of bit 1 and bit 0, modified (TC1 → SUS1, TC0 → SUS0)
		35. Camera Data Receiving Unit (CRU)	
		All	Hexadecimal notation of initial values in the table of bits, modified
		2959	(1) MIPI CSI-2: The description, modified (data → signal)
		2959	Figure 35.1 Block Diagram of CRU, modified
		3008	35.2.1 Overview: The description, modified
		3009	35.2.1.1 Features: The descriptions of Demosaicing, Statistics, and Linear Matrix, added; The descriptions of FIFO and AXI master, changed
		3010	Figure 35.8 Block Diagram of the Image Processing Module, modified
		3010	Table 35.9 Parallel Input Pins: The description of CAM_DATA15 to CAM_DATA0, modified
		3012, 3013	Table 35.10 Parallel Input Format Correspondence Table: Item “ITU-R BT.656”, added; Note 1, added
		3015, 3016	Table 35.11 Register Configuration of the Image Processing Module: Registers related to CRU Statistics, AXI-SD Master, and CRU Linear Matrix, added
		3019	35.2.3.2 CRU Interrupt Enable Register (CRUnIE): Bit 11, changed: (Reserved → SDFEOVWE); Bit 10, changed (Reserved → SDDECEE); Bit 9, changed (Reserved → SDSLVEE); Bit 8, changed (Reserved → SDFOE). The R/W attribute and description on bits 11 to 8, changed

Rev.	Date	Description	
		Page	Summary
1.40	Sep 29, 2023	3021	35.2.3.3 CRU Interrupt Status Register (CRUnINTS): Bit 11, changed (Reserved → SDFEOVWS); Bit 10, changed (Reserved → SDDECES); Bit 9, changed (Reserved → SDSLVES); Bit 8, changed (Reserved → SDFOS). The R/W attribute and description on bits 11 to 8, changed
		3024	35.2.3.7 Memory Bank 1 Base Address (Higher) Register for CRU Image Data (AMnMB1ADDRH): The description of bit 2 (reserved), modified
		3025	35.2.3.9 Memory Bank 2 Base Address (Higher) Register for CRU Image Data (AMnMB2ADDRH): The description of bit 2 (reserved), modified
		3026	35.2.3.11 Memory Bank 3 Base Address (Higher) Register for CRU Image Data (AMnMB3ADDRH): The description of bit 2 (reserved), modified
		3027	35.2.3.13 Memory Bank 4 Base Address (Higher) Register for CRU Image Data (AMnMB4ADDRH): The description of bit 2 (reserved), modified
		3028	35.2.3.15 Memory Bank 5 Base Address (Higher) Register for CRU Image Data (AMnMB5ADDRH): The description of bit 2 (reserved), modified
		3029	35.2.3.17 Memory Bank 6 Base Address (Higher) Register for CRU Image Data (AMnMB6ADDRH): The description of bit 2 (reserved), modified
		3030	35.2.3.19 Memory Bank 7 Base Address (Higher) Register for CRU Image Data (AMnMB7ADDRH): The description of bit 2 (reserved), modified
		3031	35.2.3.21 Memory Bank 8 Base Address (Higher) Register for CRU Image Data (AMnMB8ADDRH): The description of bit 2 (reserved), modified
		3040	35.2.3.32 Memory Bank 1 Base Address Lower Register for CRU Statistics Data (AMnSDMB1ADDRL), added
		3040	35.2.3.33 Memory Bank 1 Base Address Higher Register for CRU Statistics Data (AMnSDMB1ADDRH), added
		3041	35.2.3.34 Memory Bank 2 Base Address Lower Register for CRU Statistics Data (AMnSDMB2ADDRL), added
		3041	35.2.3.35 Memory Bank 2 Base Address Higher Register for CRU Statistics Data (AMnSDMB2ADDRH), added
		3042	35.2.3.36 Memory Bank 3 Base Address Lower Register for CRU Statistics Data (AMnSDMB3ADDRL), added
		3042	35.2.3.37 Memory Bank 3 Base Address Higher Register for CRU Statistics Data (AMnSDMB3ADDRH), added
		3043	35.2.3.38 Memory Bank 4 Base Address Lower Register for CRU Statistics Data (AMnSDMB4ADDRL), added
		3043	35.2.3.39 Memory Bank 4 Base Address Higher Register for CRU Statistics Data (AMnSDMB4ADDRH), added
		3044	35.2.3.40 Memory Bank 5 Base Address Lower Register for CRU Statistics Data (AMnSDMB5ADDRL), added
		3044	35.2.3.41 Memory Bank 5 Base Address Higher Register for CRU Statistics Data (AMnSDMB5ADDRH), added
		3045	35.2.3.42 Memory Bank 6 Base Address Lower Register for CRU Statistics Data (AMnSDMB6ADDRL), added
		3045	35.2.3.43 Memory Bank 6 Base Address Higher Register for CRU Statistics Data (AMnSDMB6ADDRH), added
		3046	35.2.3.44 Memory Bank 7 Base Address Lower Register for CRU Statistics Data (AMnSDMB7ADDRL), added
		3046	35.2.3.45 Memory Bank 7 Base Address Higher Register for CRU Statistics Data (AMnSDMB7ADDRH), added
		3047	35.2.3.46 Memory Bank 8 Base Address Lower Register for CRU Statistics Data (AMnSDMB8ADDRL), added
		3047	35.2.3.47 Memory Bank 8 Base Address Higher Register for CRU Statistics Data (AMnSDMB8ADDRH), added
		3048	35.2.3.48 Memory Bank Enable Register for CRU Statistics Data (AMnSDMBVALID), added

Rev.	Date	Description	
		Page	Summary
1.40	Sep 29, 2023	3049	35.2.3.49 Memory Bank Status Register for CRU Statistics Data (AMnSDMBS), added
		3050	35.2.3.50 AXI-SD Master Transfer Setting Register (AMnSDAXIATTR), added
		3051	35.2.3.51 AXI-SD Master FIFO Setting Register (AMnSDFIFO), added
		3051	35.2.3.52 AXI-SD Master Transfer Re-Start (AMnSDFIFOTRST), added
		3052	35.2.3.53 AXI-SD Master FIFO Pointer Register (AMnSDFIFOPNTR), added
		3053	35.2.3.54 AXI-SD Master Transfer Stop Register (AMnSDAXISTP), added
		3053	35.2.3.55 AXI-SD Master Transfer Stop Status Register (AMnSDAXISTPACK), added
		3056 to 3058	35.2.3.58 CRU Image Processing Main Control Register (ICnMC): Bits 25 and 24, changed (Reserved → RAWSTTYP[1:0]); The R/W attribute and description, changed. Bit 7, changed (Reserved → STITHR); The R/W attribute and description, changed. Bit 4, changed (Reserved → LMXTHR); The R/W attribute and description, changed. Bit 3, changed (Reserved → DEMTHR); The R/W attribute and description, changed. Bit 5 (CSCTHR): The description, modified.
		3059	35.2.3.58, (1) Bypass Setting for Each Function: The description, modified
		3059	Figure 35.11 Bypass Setting of Each Function, modified
		3063	35.2.3.64 CRU Linear Matrix Offset Register (ICnLMXOF), added
		3064	35.2.3.65 CRU Linear Matrix R Coefficient 1 Register (ICnLMXRC1), added
		3065	35.2.3.66 CRU Linear Matrix R Coefficient 2 Register (ICnLMXRC2), added
		3066	35.2.3.67 CRU Linear Matrix G Coefficient 1 Register (ICnLMXGC1), added
		3067	35.2.3.68 CRU Linear Matrix G Coefficient 2 Register (ICnLMXGC2), added
		3068	35.2.3.69 CRU Linear Matrix B Coefficient 1 Register (ICnLMXBC1), added
		3069	35.2.3.70 CRU Linear Matrix B Coefficient 2 Register (ICnLMXBC2), added
		3070	35.2.3.71 CRU Statistics Control 1 Register (ICnSTIC1), added
		3071	35.2.3.72 CRU Statistics Control 2 Register (ICnSTIC2), added
		3084	35.2.3.87 CRU RGB → YCbCr Color Space Conversion Y Coefficient 3 Register (ICnYCCR3): Bit 16, changed (Reserved → YCLCEN); The R/W attribute and description, changed
		3086	35.2.3.90 CRU RGB → YCbCr Color Space Conversion Cb Coefficient 3 Register (ICnCBCCR3): Bit 16, changed (Reserved → CBCLCEN); The R/W attribute and description, changed
		3088	35.2.3.93 CRU RGB → YCbCr Color Space Conversion Cr Coefficient 3 Register (ICnCRCCR3): Bit 16, changed (Reserved → CRCLCEN); The R/W attribute and description, changed
		3096 to 3098	Table 35.12 Input Format and Image Processing: The rows of functions Demosaicing, Linear Matrix, and Statistics, added
		3099, 3100	Table 35.13 Image Output Formats: The register settings of "RAW" and "MIPI CSI-2 Recommended Memory Storage Format", modified
		3101	Table 35.14 Statistics Output Formats, added
		3107	35.2.4.3, (4) Statistics Data Transfer Destination Address (Statistics Data Storage Destination), added
		3109	35.2.4.7, (1) Mode Setting, modified
		3112	35.2.4.10 Demosaicing, added
		3114	35.2.4.11 Linear Matrix, added
		3116	35.2.4.14, (1) RGB to YCbCr: The description, modified. Register Settings, modified
		3119	35.2.4.18 Statistics, added
		3132	Table 35.18 Interrupt Sources, Registers, and Interrupt Signals to the Interrupt Controller: Statistics-related interrupts and errors for axi_mst_err_int, added
		3138	Figure 35.48 Initializing the AXI Master (MIPI CSI-2 Input), modified
		3139, 3140	Figure 35.49 Initializing Image Processing (MIPI CSI-2 Input), modified
		3146	Figure 35.55 Stopping Image Processing Reception (MIPI CSI-2 Input), modified

Rev.	Date	Description	
		Page	Summary
1.40	Sep 29, 2023	3147	35.3.3 Frame Synchronization Settings: The description, modified
		3147	Figure 35.56 Flow for Changing the Frame Synchronization Settings (by Setting Registers), modified
		3148	Figure 35.57 Flow for Changing the Frame Synchronization Settings (by Using the DMAC), modified
1.45	Jan 31, 2024	34. MIPI DSI	
		2840, 2841	Table 34.3 Resister list (2/3), (3/3): The access size of the SQCHxDSCyAR, SQCHxDSCyBR, SQCHxDSCyCR, and SQCHxDSCyD registers, modified
		2912	34.3.1.59 Sequence Channel x Descriptor y-A Register (SQCHxDSCyAR): The description on the reserved bits (bit 31, bit 30), modified
		2914	34.3.1.60 Sequence Channel x Descriptor y-B Register (SQCHxDSCyBR): The description on the reserved bits (bit 31 to bit 26, bit 23 to bit 0), modified
		2915	34.3.1.61 Sequence Channel x Descriptor y-C Register (SQCHxDSCyCR): The description on the reserved bits (bit 23, bit 21 to bit 0), modified
		2933	Figure 34.9 Single Packet Transfer: The description of the SQCHxDSCyBR.STT bit and the SQCHxDSCyBR.REPEAT bit, deleted
		2935	Figure 34.10 Single Packet Receive: The description of the SQCHxDSCyBR.STT bit and the SQCHxDSCyBR.REPEAT bit, deleted
		2936	Figure 34.11 Trigger transmission: The description of the SQCHxDSCyBR.STT bit and the SQCHxDSCyBR.REPEAT bit, deleted

RZ/G2L Group, RZ/G2LC Group User's Manual: Hardware

Publication Date: Rev.1.00 Sep 7, 2021
Rev.1.45 Jan 31, 2024

Published by: Renesas Electronics Corporation

RZ/G2L Group, RZ/G2LC Group

Renesas Electronics Corporation